US006160818A

United States Patent [19]

Berger et al.

[11] Patent Number: 6,160,818

[45] Date of Patent: *Dec. 12, 2000

[54] **TRAFFIC MANAGEMENT IN PACKET COMMUNICATION NETWORKS HAVING SERVICE PRIORITIES AND EMPLOYING EFFECTIVE BANDWIDTHS**

[75] Inventors: **Arthur W. Berger**, Fair Haven; **Ward Whitt**, Basking Ridge, both of N.J.

[73] Assignee: **AT &T Corp**, New York, N.Y.

[*] Notice: This patent issued on a continued prosecution application filed under 37 CFR 1.53(d), and is subject to the twenty year patent term provisions of 35 U.S.C. 154(a)(2).

[21] Appl. No.: **08/895,641**

[22] Filed: **Jul. 17, 1997**

[51] **Int. Cl.**[7] **H04J 3/16**; G01R 31/08

[52] **U.S. Cl.** **370/468**; 370/232; 370/233; 370/230

[58] **Field of Search** 370/230, 231, 370/232, 235, 236, 237, 412, 462, 468, 477, 428

[56] **References Cited**

U.S. PATENT DOCUMENTS

5,040,176	8/1991	Barzilai	370/422
5,121,383	6/1992	Golestani	370/235
5,233,604	8/1993	Ahmadi	370/238
5,267,232	11/1993	Katsube	370/230
5,289,462	2/1994	Ahmadi et al.	370/232
5,309,433	5/1994	Cidon	370/390
5,311,513	5/1994	Ahmadi	370/230
5,313,454	5/1994	Bustini	370/231
5,347,511	9/1994	Gun	370/255
5,359,593	10/1994	Derby	370/234
5,367,523	11/1994	Chang	370/235
5,408,465	4/1995	Gusella	370/231
5,434,848	7/1995	Chimento	370/232
5,521,971	5/1996	Key et al. .	
5,687,167	11/1997	Bertin	370/254
5,719,854	2/1998	Choudhury	370/231
5,781,624	7/1998	Mitra	379/244
5,790,522	8/1998	Fichou	370/236
5,828,653	10/1998	Goss	370/230
5,838,663	11/1998	Elwalid	370/233

OTHER PUBLICATIONS

R. Bolla, et al., "Adaptive Access Control of Multiple Traffic Classes in ATM Networks", Globecom '91.

Primary Examiner—Douglas W. Olms
Assistant Examiner—Ricardo M. Pizarro

[57] **ABSTRACT**

A method is provided for admitting new requests for service in a shared resource having a capacity. The new request has service priority levels associated therewith. In one embodiment of the invention, for example, the shared resource may be a packet communications network and the service request may be a request to admit a new connection. The method proceeds as follows. First, for each service priority level on said shared resource, a total effective bandwidth is generated which is represented by a sum of individual effective bandwidths of previously admitted requests for service. Subsequent to receiving a new request for service having a specified priority of service level, a plurality of effective bandwidths are accessed for the new request. The plurality of effective bandwidths are respectively associated with the specified service priority level and service priority levels therebelow. The new request is admitted if, for the specified service priority level and for each service priority level therebelow, the sum of (i) said total effective bandwidth for a given service priority level and (ii) for said new request, the effective bandwidth at the given service priority is less than the capacity.

29 Claims, 4 Drawing Sheets

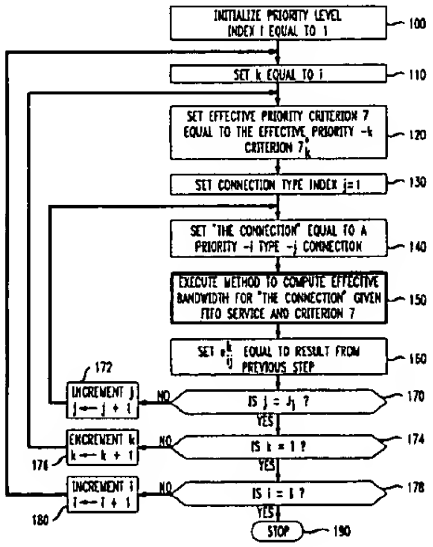


FIG. 1

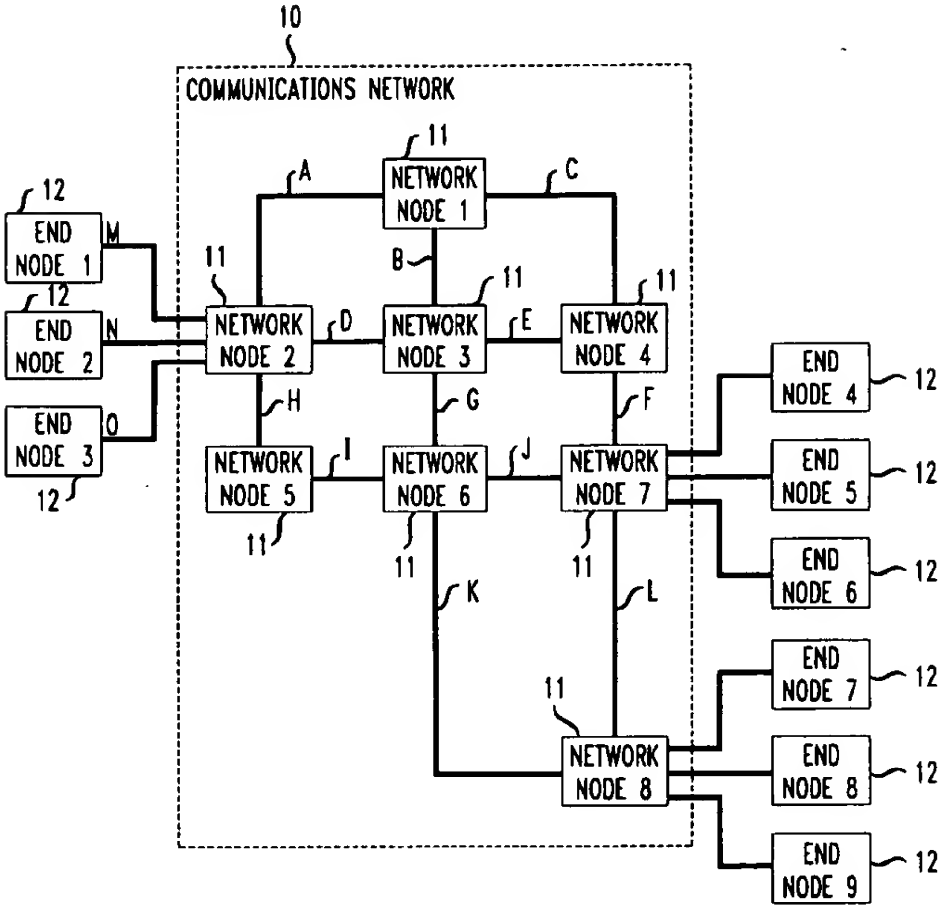

10
COMMUNICATIONS NETWORK
11
NETWORK NODE 1
A
C
B
12
END NODE 1
M
11
NETWORK NODE 2
D
11
NETWORK NODE 3
E
11
NETWORK NODE 4
12
END NODE 2
N
END NODE 4
12
H
G
F
END NODE 3
O
I
J
END NODE 5
12
NETWORK NODE 5
NETWORK NODE 6
NETWORK NODE 7
12
11
11
11
END NODE 6
12
K
L
END NODE 7
12
11
NETWORK NODE 8
END NODE 8
12
END NODE 9
12

FIG. 2

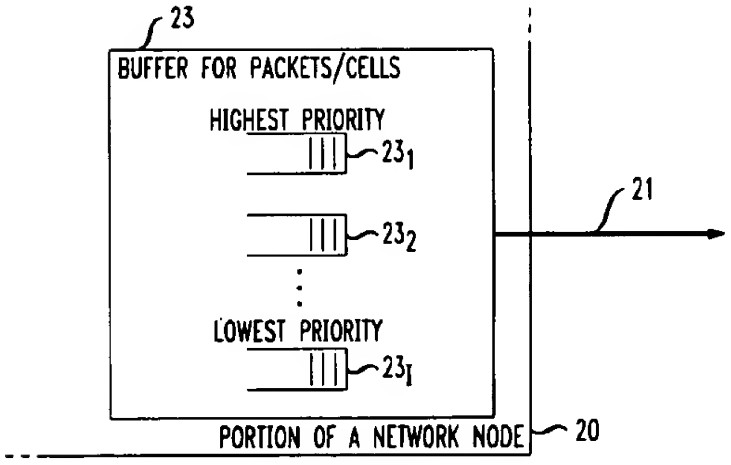

23
BUFFER FOR PACKETS/CELLS
HIGHEST PRIORITY
23_1
23_2
21
LOWEST PRIORITY
23_I
PORTION OF A NETWORK NODE
20

FIG. 3

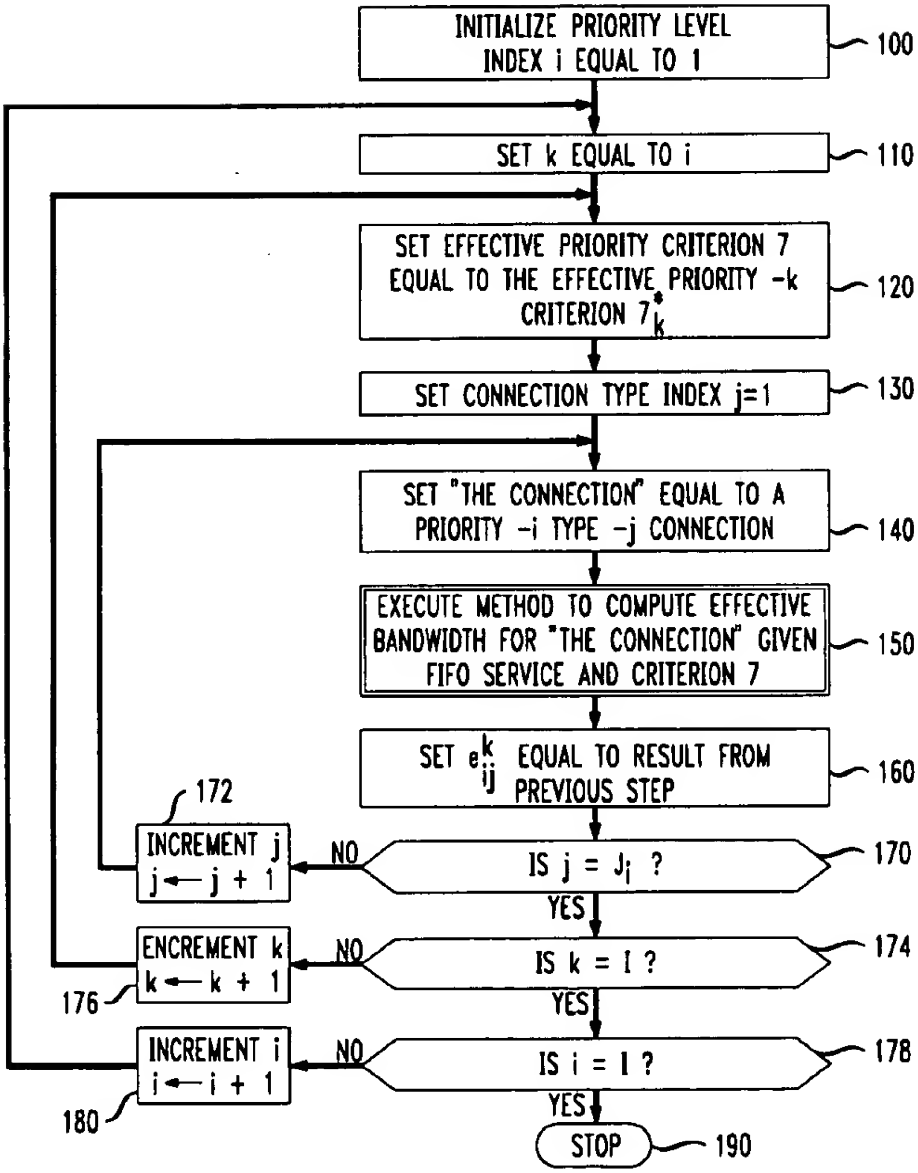

INITIALIZE PRIORITY LEVEL INDEX i EQUAL TO 1
100
SET k EQUAL TO i
110
SET EFFECTIVE PRIORITY CRITERION 7 EQUAL TO THE EFFECTIVE PRIORITY -k CRITERION 7*k
120
SET CONNECTION TYPE INDEX j=1
130
SET "THE CONNECTION" EQUAL TO A PRIORITY -i TYPE -j CONNECTION
140
EXECUTE METHOD TO COMPUTE EFFECTIVE BANDWIDTH FOR "THE CONNECTION" GIVEN FIFO SERVICE AND CRITERION 7
150
SET e^k_ij EQUAL TO RESULT FROM PREVIOUS STEP
160
IS j = Jᵢ ?
170
NO
YES
172
INCREMENT j j ← j + 1
IS k = I ?
174
NO
YES
ENCREMENT k k ← k + 1
176
IS i = I ?
178
NO
YES
INCREMENT i i ← i + 1
180
STOP
190

FIG. 4

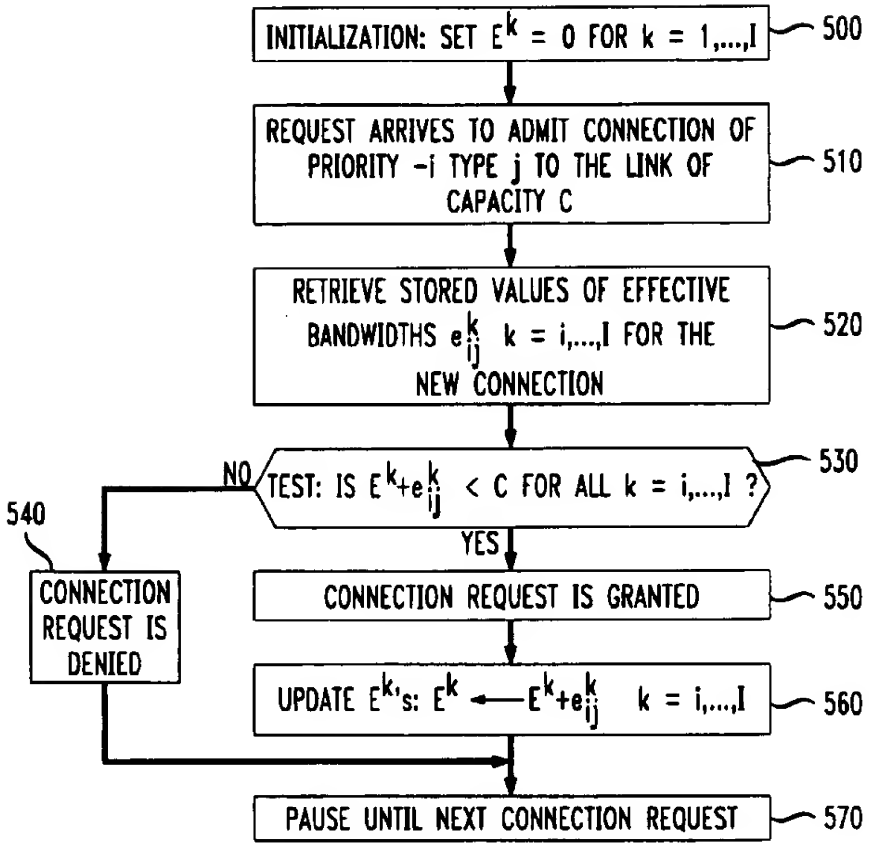

INITIALIZATION: SET $E^k = 0$ FOR k = 1,...,I
500
REQUEST ARRIVES TO ADMIT CONNECTION OF PRIORITY -i TYPE j TO THE LINK OF CAPACITY C
510
RETRIEVE STORED VALUES OF EFFECTIVE BANDWIDTHS e_{ij}^k k = i,...,I FOR THE NEW CONNECTION
520
TEST: IS $E^k+e_{ij}^k < C$ FOR ALL k = i,...,I ?
530
NO
YES
540
CONNECTION REQUEST IS DENIED
CONNECTION REQUEST IS GRANTED
550
UPDATE E^k's: $E^k \leftarrow E^k+e_{ij}^k$ k = i,...,I
560
PAUSE UNTIL NEXT CONNECTION REQUEST
570

FIG. 5

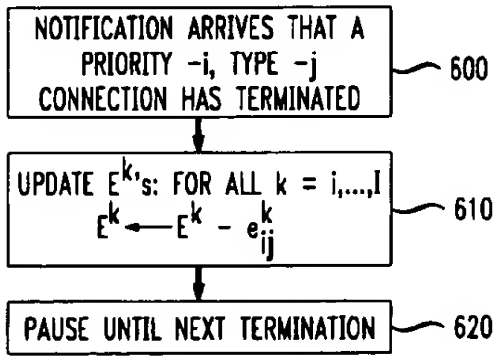
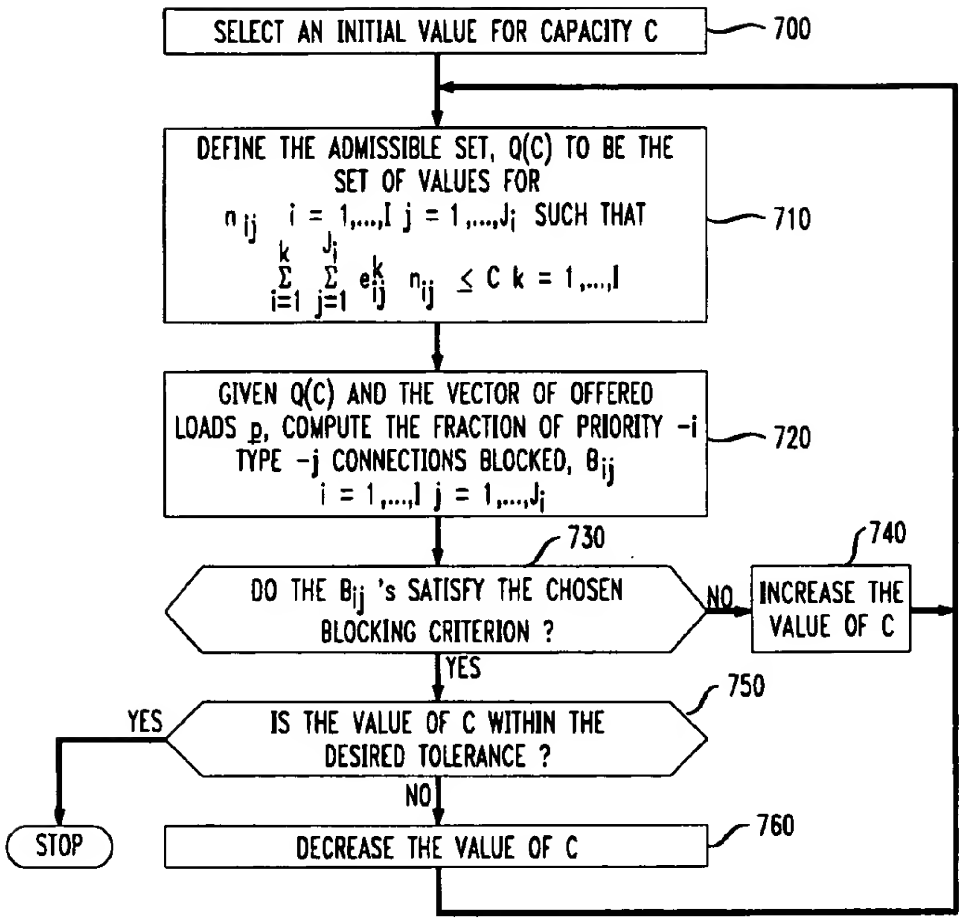

NOTIFICATION ARRIVES THAT A PRIORITY -i, TYPE -j CONNECTION HAS TERMINATED
600
UPDATE E^k's: FOR ALL k = i,...,I $E^k \leftarrow E^k - e_{ij}^k$
610
PAUSE UNTIL NEXT TERMINATION
620

FIG. 6


SELECT AN INITIAL VALUE FOR CAPACITY C
700
DEFINE THE ADMISSIBLE SET, Q(C) TO BE THE SET OF VALUES FOR n_{ij} $i = 1,...,I$ $j = 1,...,J_i$ SUCH THAT $\sum_{i=1}^{k} \sum_{j=1}^{J_i} e_{ij}^{k} n_{ij} \leq C$ $k = 1,...,I$
710
GIVEN Q(C) AND THE VECTOR OF OFFERED LOADS p, COMPUTE THE FRACTION OF PRIORITY -i TYPE -j CONNECTIONS BLOCKED, B_{ij} $i = 1,...,I$ $j = 1,...,J_i$
720
730
DO THE B_{ij} 's SATISFY THE CHOSEN BLOCKING CRITERION ?
NO
740
INCREASE THE VALUE OF C
YES
750
IS THE VALUE OF C WITHIN THE DESIRED TOLERANCE ?
YES
STOP
NO
DECREASE THE VALUE OF C
760

TRAFFIC MANAGEMENT IN PACKET COMMUNICATION NETWORKS HAVING SERVICE PRIORITIES AND EMPLOYING EFFECTIVE BANDWIDTHS

BACKGROUND OF THE INVENTION

Emerging high-speed communication networks, such as broadband ISDN networks that employ ATM technology, tend to be packet networks rather than circuit-switched networks because the packet structure allows for better resource sharing. In a packet network, sources do not require dedicated bandwidth (e.g., circuits) for the entire duration of a connection. Unfortunately, however, the enhanced flexibility of packet networks also makes it more difficult to effectively control the admission of connections seeking to enter an existing network and to plan the capacity of future networks when they are designed.

The problems of admission control and capacity planning in a packet network may be addressed by a concept known as the "effective bandwidth" or "equivalent bandwidth" of a connection. When employing this concept, an appropriate effective bandwidth is assigned to each connection and each connection is treated as if it required this effective bandwidth throughout the active period of the connection. The feasibility of admitting a given set of connections may then be determined by ensuring that the sum of the effective bandwidths is less than or equal to the total available bandwidth (i.e., the capacity). By using effective bandwidths in this manner, the problems of admission control and capacity planning are addressed in a fashion similar to that employed in circuit-switched networks. Additional details concerning effective bandwidths may be found in U.S. Pat. Nos. 5,289,462 and 5,521,971, for example.

It is often desirable to provide different quality of service (QOS) guarantees to different classes of customers who use a communication network. Network nodes have been designed that partition the connections established on a link into different priority levels, whereby all of the packets queued from connections of a given priority are emitted prior to any packets from connections of a lower priority. Accordingly, allocation of network resources based on different priority levels is becoming a realistic possibility. Known admission control methods which use the effective bandwidth concept, however, assume that the network nodes operate on a First-In-First-Out (FIFO) basis.

It is therefore desirable to employ effective bandwidths for admission control and capacity planning which can account for priorities of service at the network nodes.

SUMMARY OF THE INVENTION

In accordance with the present invention, a method is provided for admitting new requests for service in a shared resource having a capacity. The new request has service priority levels associated therewith. In one embodiment of the invention, for example, the shared resource may be a packet communications network and the service request may be a request to admit a new connection. The method proceeds as follows. First, for each service priority level on said shared resource, a total effective bandwidth is generated which is represented by a sum of individual effective bandwidths of previously admitted requests for service. Subsequent to receiving a new request for service having a specified priority of service level, a plurality of effective bandwidths are accessed for the new request. The plurality of effective bandwidths are respectively associated with the specified service priority level and service priority levels

therebelow. The new request is admitted if, for the specified service priority level and for each service priority level therebelow, the sum of (i) said total effective bandwidth for a given service priority level and (ii) for said new request, the effective bandwidth at the given service priority is less than the capacity.

Various embodiments of the invention also encompass methods for releasing a request for service and a method for dimensioning the capacity of the shared resource.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** shows a simplified block diagram of a packet transmission system.

FIG. **2** shows a block diagram of a network node configured to supply packets to an outgoing link based on priority levels that are assigned to each connection.

FIG. **3** shows a flow chart of an exemplary method for calculating effective bandwidths in accordance with the present invention.

FIG. **4** shows a flowchart of an exemplary method for admitting a connection in accordance with the present invention.

FIG. **5** shows a flowchart of an exemplary method for releasing allocated bandwidth upon termination of a connection in accordance with the present invention.

FIG. **6** shows a flowchart of an exemplary method for dimensioning a network in accordance with the present invention.

DETAILED DESCRIPTION

Referring more particularly to FIG. **1**, there is shown a general block diagram of a packet transmission system **10** comprising eight network nodes **11** numbered **1** through **8**. Each of network nodes **11** is linked to others of the network nodes **11** by one or more communication links A through L. Each such communication link can support multiple connections, which in turn may be either semi-permanent connections or selectively enabled (dial-up) connections. Any or all of network nodes **11** may be attached to end nodes, network node **2** being shown as attached to end nodes **1, 2** and **3**, network node **7** being shown as attached to end nodes **4, 5** and **6**, and network node **8** being shown as attached to end nodes **7, 8** and **9**. Network nodes **11** each comprise a data processing system which provides data communications services to all connected nodes, network nodes and end nodes, as well as decision points within the node. The network nodes **11** each comprise one or more decision points within the node, at which incoming data packets are selectively routed on one or more of the outgoing communication links to another node. Such routing decisions are made in response to information in the header of the data packet. The network node also provides ancillary services such as the calculation of routes or paths between terminal nodes, providing access control to packets entering the network at that node, and providing directory services and maintenance of network topology data bases used to support route calculations.

The steps involved in establishing a connection between two nodes in transmission system **10** include bandwidth allocation, path selection, admission control, and call setup. Bandwidth allocation and admission control are accomplished by determining the "effective bandwidth" of the connection to be established. The effective bandwidth of a connection assesses the loading that the new connection will generate, based on the traffic characteristics of the source

and the desired quality of service. The transmission system assumes that the connection will require the effective bandwidth throughout its duration.

U.S. Pat. No. 5,289,462 provides one method that may be used to calculate the effective bandwidth of a connection (see equation 2 of the patent). However, as previously noted, this known technique does not take into account different priority levels that may be assigned to the connections. Rather, this reference simply employs FIFO service.

FIG. 2 shows a block diagram of a network node **20** configured to supply packets to an outgoing link **21** based on the assigned priority level of each connection. The node **20** includes a buffer **23** for queuing the connections. Queue 1 in buffer **23** has the highest priority and packets are supplied from this queue onto link **21** in a FIFO order until the queue is empty. When queue 1 is empty, packets are supplied from queue 2 in a FIFO order as long as queue 1 remains empty. The process continues until the lowest priority queue (queue 1) is reached, which will occur when all higher priority queues have been emptied. One of ordinary skill in the art will recognize that the queues 1, 2, . . . , I may be physically realized in a single buffer or in separate buffers. In current packet switches typical implementations of service priorities have two to four priority levels (I would equal 2 to 4), though the invention pertains for an arbitrary number of priority levels.

To account for service priorities, the invention assigns each connection (other than those in the lowest priority) multiple effective bandwidths: one for the priority-of-service level of the given connection and one for each lower priority level. The individual effective bandwidths can be calculated in accordance with known procedures employing a FIFO discipline. For purposes of illustration, the present invention will be described in what follows for the case of a single resource. One of ordinary skill in the art will recognize that the invention also encompasses situations involving two or more resources.

Prior art applied to a node with service priorities would assign a single effective bandwidth to each connection. In particular, let e_{ij} represent the effective bandwidth required by a connection having a priority i (lower indices have higher priority so that priority 1 has the highest priority) and characteristics of a source type j, and let n_{ij} be the number of connections of priority i and source type j. The number of priority levels is denoted by I. The number of source types having priority i is denoted by J_i. If the capacity of a link is denoted by C, then the following inequality constraint would be used by prior art:

$$\sum_{i=1}^{I}\sum_{j=1}^{J_i} e_{ij}n_{ij} \le C \qquad (1)$$

The set of numbers n_{ij} of connections satisfying this inequality is referred to as the admissible set.

The present inventors have recognized that inequality constraint (1) is unnecessarily conservative. A significant improvement can be obtained by, first, having a separate inequality constraint for each priority level and, second, by assigning different effective bandwidths to the same connection to be used in the different inequality constraints. That is, the single inequality constraint (1) should be replaced by the set of I inequality constraints:

$$\sum_{i=1}^{k}\sum_{j=1}^{J_i} e_{ij}^{k}n_{ij} < C \quad \text{for each priority level } k,\ k = 1, \ldots, I \qquad (2)$$

where e_{ij}^{k} is the effective bandwidth of an (i,j) connection (that is, a connection of source type j and priority level i) as seen by priority k, and C is the capacity of the link. In other words, a given connection may be assigned several effective bandwidths—one effective bandwidth for the priority of service level of the given connection and one effective bandwidth for each lower priority level. It should be noted that for some service types distinct effective bandwidths for all lower priorities may yield only modest efficiency gains, in which case to reduce complexity, for a given (i, j) connection the values of the effective bandwidth e_{ij}^{k} may be the same for different values of the priority k. The admissible set associated with (2) is the set of numbers n_{ij} of connections satisfying all I inequalities in (2). At first glance, the I inequality constraints in (2) might seem more restrictive than the single inequality constraint in (1). However, if $e_{ij}^{k} \cong e_{ij}$ for all i, j, and $k \cong i$, where e_{ij}^{k} is in (2) and e_{ij} is in (1), as should typically be the case in practice, then the new admissible set in (2) is at least as large as the admissible set in (1). In fact, the inventors have shown that the admissible set for (2) can be much larger than the admissible set for (1).

The previously described admissible sets apply to the case of a single resource (e.g., link). However, a connection may require multiple resources. In such cases the connection must satisfy the inequality constraints associated with all its required resources.

The effective bandwidth for FIFO service such as discussed in the previously mentioned patent does not depend on whether the performance criterion is specified in terms of loss or delay. Loss is typically modeled in terms of the probability that the queue length is greater than a specified threshold in an infinite-capacity buffer. When the concept of priorities is employed, however, the effective bandwidth does depend on whether the performance criterion is based on loss or delay. For all but the highest priorities, the work-in-system (which represents the queue length for constant size packets) can be significantly smaller than the delay because the processing of work can be interrupted by arrivals from a higher priority connection. The present invention will be illustrated with a performance criterion based on loss. However, one of ordinary skill in the art will recognize that the present invention may be adapted to situations in which the performance criterion is based on delay.

The required effective bandwidths e_{ij}^{k} can be determined by a variety of methods. In accordance with one embodiment of the present invention, an approach based on large buffer asymptotics will now be described. For purposes of the following discussion, the performance criterion is modeled on the workload in a hypothetical infinite-capacity system being above a threshold b_i with a probability no more than p_i, where b_i represents the capacity of the buffer for priority i in the node. That is, the priority-i criterion based on loss may be represented by:

$$\text{Probability (priority-i workload} > b_i) < p_i,\ i = 1, \ldots I \qquad (3)$$

Based on simplifying assumptions, the effective bandwidth satisfying this loss criterion may be expressed as follows:

$$e_{ij}^{k} = \begin{cases} 0, & i > k \\ \psi_{A_{ij}}(\eta_k^*)/\eta_k^*, & i \le k \end{cases} \quad (4)$$

where n^*_k is called the effective priority-k criterion and is set equal to $-\log(p_k)/b_k$ and

$$\psi_{A_{ij}}(\theta) = \lim_{t\to\infty} t^{-1}\log E e^{\theta A_{ij}(t)} \quad (5)$$

and A_{ij} (t) is the input of work of an (i,j) connection during the interval [0, t]. Equation (4) will be referred to as the empty-buffer effective bandwidth (EBEB) approximation. It should be emphasized that (3)–(5) provides only one way to obtain the required effective bandwidth. The present invention contemplates the use of alternative methods as well.

FIG. **3** shows an example of the inventive method for determining the effective bandwidth given different priorities of service. The results of this method, which yield a series of numerical values for e_{ij}^{k}, may be calculated offline and stored for subsequent retrieval on an as-needed basis. As shown in the figure, the priority level index i is initialized to one (step **100**), and the index k is set equal to i (step **110**). The effective priority criterion η is set equal to the effective priority-k criterion , i.e., $\eta=\eta^*_k$ (step **120**), and the index j denoting the connection type is set equal to one (step **130**). The particular effective bandwidth to be calculated is for connection (i,j)as seen by priority k (step **140**). The effective bandwidth e^{ijk} is calculated in accordance with known techniques employing FIFO service, such as shown by equation 2 in U.S. Pat. No. 5,289,462, for example (steps **150** and **160**). The process continues through steps **170–180** by repeating steps **110–160** until values of the effective bandwidth e_{ij}^{k} have been calculated for all values of i,j and k.

FIG. **4** shows a flowchart of an exemplary method for admittance of connections on a link from a network node. The method employs the effective bandwidths that have been calculated by the method of FIG. **3**. A value E_k is defined, which equals the sum of the effective bandwidths of connections that are currently admitted to the network with a priority k. The value of E_k is initially set to zero. (step **500**). Upon receiving a request to admit a connection of priority i, type j, (step **510**), the previously determined values of the effective bandwidths e_{ij}^{k}, k=i, . . . I, for this connection is retrieved from a database (step **520**). The connection will be admitted to the network if the following (I–i+1) inequalities are satisfied: $E_k+e_{ij}^{k}<C$ for all k=i, . . . , I (step **530**). If the inequalities are not satisfied, the connection request is denied (step **540**). If the inequalities are satisfied, the connection request is granted (step **550**) and the value of E_k is updated for k=i, . . . , I to the reflect the newly admitted connection (step **560**). The process then exits until a new connection request is received (step **570**).

FIG. **5** shows a flowchart of an exemplary method for releasing allocated bandwidth on a link upon termination of a connection. Upon receiving notification that connection ij has terminated (step **600**), the value of E_k is updated, for priority classes k=i, . . . I, by subtracting for each k=i, . . . , I, the value of the effective bandwidth, e_{ij}^{k}, for the released connection from the current value of E_k (step **610**). The process exits and resumes when a subsequent connection is terminated.

The methods described in FIGS. **3–5** assume that the effective bandwidths have been calculated offline and remain fixed. Alternatively, however, the effective bandwidths may be adjusted based on system measurements while the call is in progress. Such effective bandwidth measurements can be accomplished in a variety of ways. For example, the value of $\psi_{A_{ij}}(\theta)$ in (5) may be estimated by the observed value:

$$\psi_{A_{ij}}(\theta)=t^{-1}e^{\theta A_{ij}(t)} \quad (6)$$

for suitably large t. To apply (4), this estimation would need for performed for $\theta=\eta^*_k$ for all k with $i\leqq k\leqq I$.

FIG. **6** shows a flowchart of an exemplary method for dimensioning a link in a network given an admissible set of numbers n_{ij} of connections. The process begins by selecting an initial value for the capacity (step **700**), which is subsequently iterated until a final value is reached. More specifically, the admissible set $\alpha(C)$ is the set of values n_{ij} such that the set of inequalities are satisfied for the selected capacity (step **710**). Given $\alpha(C)$ and the vector of offered loads ρ (i.e., the $(i,j)^{th}$ offered load, ρ_{ij}, is the product of the arrival rate of requests for connections (i,j) and the mean holding-time of these connections), the fraction of connections (i, j) that would be blocked given this capacity is determined in accordance with known methods such as disclosed in G. Choudhury et al., *Advances in Applied Probability*, Vol. 27, 1995, pp. 1104–1143, for example (step **720**). This fraction of blocked connections is measured against a predetermined blocking criterion that is provided as an input parameter (step **730**). A typical blocking criterion that may be employed is that less than 1% of the connections (i,j) are blocked for each value i and j. If the blocking criterion is not satisfied, the value of the capacity C is increased (step **740**) and the process begins again by defining an admissible set that satisfies inequality (2) for the newly selected capacity. Alternatively, if the blocking criterion is satisfied, the method determines whether the capacity is within a desired tolerance range (step **750**). If the capacity is within the desired tolerance, the method is complete. If the capacity is not within the desired tolerance, the capacity is decreased and the method repeats for the new value of the capacity.

The present inventors have recognized that set of I constraints in (2) for a single link based on I priority levels is equivalent to the set of I constraints associated with a network of I links using the FIFO service discipline. Thus, previous methods for dimensioning FIFO networks, such as shown in the Choudhury reference mentioned above, can be applied to priority networks. With priority networks, there are I constraints for each resource (e.g., link) in the network.

As previously mentioned, the effective bandwidths may be determined in accordance with a variety of methods such as the EBEB approximation of equation (4). However, other methods may be more applicable in certain circumstances. For example, if the network employs Asynchronous Transfer Mode (ATM) technology that uses the Statistical Bit Rate (SBR) transfer capability, the effective bandwidths may be determined in terms of conventional traffic descriptors such as the Peak Cell Rate (PCR) and the Sustainable Cell Rate (SCR) (see ITU Recommendation 1.371 'B-ISDN Traffic Control and Congestion Control', Geneva, May, 1996). In this case, a connection (i,j) can be assigned an effective bandwidth e_{ij}^{k} for priority level k, $k\geqq i$, that is some function of the PCR's and SCR's of all existing connections and the candidate new connection with priority $k\geqq i$. For example, e_{ij}^{i} might be set equal to its PCR (denoted PCR_{ij}) for the connection's priority level i, while for k>i, e_{ij}^{k} might be set equal to α_{ij}^{k} (SCR_{ij}) for the lower priorities k=i+1, . . . , I, where α_{ij}^{k} is a selected positive number.

One of ordinary skill in the art will recognize that the applicability of present invention is not limited to packet

communications networks. Rather, the invention is equally applicable to other resource sharing systems in which a priority status is assigned to something other than a data packet.

What is claimed is:

1. A method for admitting a new connection having a given service priority level in a packet communications network that includes a plurality of constrained resources including switching nodes and transmission links, each having a capacity, said method comprising the steps of:

generating, for each constrained resource that may be selected to effect said new connection, and for each service priority level, a total effective bandwidth represented by a sum of individual effective bandwidths of previously admitted connections;

determining a plurality of effective bandwidths for said new connection, said plurality of effective bandwidths respectively being associated with said given service priority level and lower service priority levels, and being based on a loss criterion;

admitting said new connection if constrained resources can be selected from among those of said constrained resources for which said step of generating developed said total effective bandwidth, such that for each of the selected resources, for said given priority level of said new connection and for all lower priority levels, the sum of said total effective bandwidth and said determined effective bandwidth is less than said capacity.

2. The method of claim 1 wherein said plurality of effective bandwidths of said new connection is based on a network end-to-end delay criterion.

3. The method of claim 1 wherein said plurality of effective bandwidths of said new connection is determined in accordance with an empty-buffer effective bandwidth (EBEB) approximation.

4. The method of claim 1 wherein said determining step includes the step of

retrieving from a database previously calculated values of said plurality of effective bandwidths of said new connection.

5. The method of claim 1 wherein said determining step includes the step of obtaining measured values for said plurality of effective bandwidths of said new connection.

6. The method of claim 1 wherein a plurality of said effective bandwidths of at least one type of a new connection have the same value.

7. The method of claim 1 wherein admitted connections and said new connection are classified into source-types such that connections of a given source-type and a given service priority level have the same respective values for said plurality of effective bandwidths.

8. The method of claim 1 wherein at least one of said transmission links is defined in terms of the portion of bandwidth on a transmission path that is dedicated to said connections.

9. A method for updating bandwidths upon releasing a connection in a packet communications network that includes a plurality of switching nodes interconnected by transmission links having a capacity, said method comprising the steps of:

receiving a request to terminate a connection having a specified level of service priority;

for said specified level of service priority and for each service priority level therebelow, accessing a total effective bandwidth represented by a sum of effective bandwidths of previously admitted connections,

accessing a plurality of effective bandwidths for said connection to be terminated, said plurality of effective bandwidths respectively representing said specified level of service priority and service priority levels therebelow;

for said specified level of service priority and for each service priority level therebelow, generating an updated total effective bandwidth by subtracting, for said connection to be terminated, the effective bandwidth for the given service priority level; and

releasing said connection to be terminated

wherein said plurality of effective bandwidths of said new connection is based on a loss criterion.

10. The method of claim 9 wherein said plurality of effective bandwidths of said new connection is based on a delay criterion.

11. The method of claim 9 wherein said plurality of effective bandwidths of said new connection is determined in accordance with an empty-buffer effective bandwidth approximation.

12. The method of claim 9 wherein said accessing step includes the step of retrieving from a database previously calculated values of said plurality of effective bandwidths of said new connection.

13. The method of claim 9 wherein admitted connections and said released connection are classified into source-types such that connections of a given source-type and a given service priority level have the same values for said plurality of effective bandwidths.

14. The method of claim 9 wherein at least one of said transmission links is defined in terms of the portion of bandwidth on a transmission path that is dedicated to said connections.

15. A method for dimensioning a link in a packet communications network given: (i) an initial value for capacity of the link, (ii) a classification of possible connections into a given specified level of service priority and source type, (iii) a plurality of effective bandwidths for each said classified connection, said plurality of effective bandwidths respectively being associated with said connection's service priority level and service priority levels therebelow (iv) an offered load vector for said classification of connections, and (v) a blocking criterion, said method comprising the steps of:

a. for the capacity of said link and the said plurality of effective bandwidths for each said classified connection, determining an admissible set of said connections;

b. determining the fraction of connections that are blocked for each service priority level and each source type, given said admissible set and said offered load vector;

c. repeating steps (a)–(b) wherein said link capacity is increased if said fraction of blocked connections exceeds said blocking criterion and said link capacity is decreased if said blocking criterion is satisfied until an incremental decrease in said capacity would cause said blocking criterion to be violated.

16. The method of claim 15 wherein said fraction of connections determined in step (b) is determined in accordance with an algorithm employed in a FIFO network.

17. The method of claim 10 wherein said link is defined in terms of the portion of bandwidth on a transmission path to be dedicated to the said connections.

18. A method for admitting new requests for service in a shared resource having a capacity, said new requests having service priority levels, said method comprising the steps of:

generating, for each service priority level on said shared resource, a total effective bandwidth represented by a sum of individual effective bandwidths of previously admitted requests for service;

subsequent to receiving a new request for service having a specified priority of service level, accessing a plurality of effective bandwidths for said new request, said plurality of effective bandwidths respectively being associated with said specified service priority level and service priority levels therebelow; and

admitting said new request if, for said specified service priority level and for each said service priority level therebelow, the sum of (i) said total effective bandwidth for a given service priority level and (ii) for said new request, the said effective bandwidth at said given service priority is less than said capacity

wherein said plurality of effective bandwidths of said new request for service is based on a loss criterion.

19. The method of claim **18** wherein said plurality of effective bandwidths of said new request for service is based on a delay criterion.

20. The method of claim **18** wherein said plurality of effective bandwidths of said new request for service is determined in accordance with an empty-buffer effective bandwidth (EBEB) approximation.

21. The method of claim **18** wherein said accessing step includes the step of retrieving from a database previously calculated values of said plurality of effective bandwidths of said new request for service.

22. The method of claim **18** wherein said accessing step includes the step of obtaining measured values for said plurality of effective bandwidths of said new request for service.

23. The method of claim **18** wherein said shared resource is a packet communications network.

24. A method for updating occupancy of a shared resource upon releasing a service having a specified level of service priority, in the shared resource having a capacity, said method comprising the steps of:

receiving a request to terminate said service;

for said specified level of service priority and for each service priority level therebelow, accessing a total effective bandwidth represented by a sum of effective bandwidths of previously admitted services, and

accessing a plurality of effective bandwidths for said service to be terminated, said plurality of effective bandwidths respectively representing said specified level of service priority and service priority levels therebelow;

generating an updated total effective bandwidth by subtracting from said sum said plurality of effective bandwidths accessed in said step of accessing; and

releasing said service to be terminated

wherein said plurality of effective bandwidths of said service is based on a loss criterion.

25. The method of claim **24** wherein said plurality of effective bandwidths of said service is based on a delay criterion.

26. The method of claim **24** wherein said plurality of effective bandwidths of said service is determined in accordance with an empty-buffer effective bandwidth approximation.

27. The method of claim **24** wherein said accessing step includes the step of retrieving from a database previously calculated values of said plurality of effective bandwidths of said service.

28. A method for dimensioning a shared resource given: (i) an initial value for capacity of the shared source, (ii) a classification of service requests into a given specified level of service priority and source type, (iii) a plurality of effective bandwidths for each of said service requests, where said plurality of effective bandwidths for a service request respectively being associated with a service priority level of said service request and service priority levels lower than the service priority level of said service request (iv) an offered load vector for said classification of service requests, and (v) a blocking criterion, said method comprising the steps of:

a. for said capacity and said plurality of effective bandwidths for each of said classified service requests, determining an admissible set of said connections;

b. determining a fraction of service requests that are blocked for each service priority level and each source type, given said admissible set and said offered load vector;

c. repeating steps (a)–(b) wherein said capacity is increased if said fraction of blocked requests exceeds said blocking criterion and said capacity is decreased if said blocking criterion is satisfied until an incremental decrease in said capacity would cause said blocking criterion to be violated.

29. The method of claim **28** wherein said fraction of service requests determined in step (b) is determined in accordance with an algorithm employed in a FIFO network.

* * * * *

US006021263A

United States Patent [19]

Kujoory et al.

[11] Patent Number: 6,021,263

[45] Date of Patent: *Feb. 1, 2000

[54] **MANAGEMENT OF ATM VIRTUAL CIRCUITS WITH RESOURCES RESERVATION PROTOCOL**

[75] Inventors: **Ali Mohammad Kujoory**, Lincroft; **Samir S. Saad**, Long Branch; **David Hilton Shur**, Middletown; **Kamlesh T. Tewani**, Freehold; **James Kwong Yee**, Marlboro, all of N.J.

[73] Assignee: **Lucent Technologies, Inc.**, Murray Hill, N.J.

[*] Notice: This patent issued on a continued prosecution application filed under 37 CFR 1.53(d), and is subject to the twenty year patent term provisions of 35 U.S.C. 154(a)(2).

[21] Appl. No.: **08/602,428**

[22] Filed: **Feb. 16, 1996**

[51] **Int. Cl.**[7] **G06F 13/33**; G06F 15/17

[52] **U.S. Cl.** **395/200.62**; 395/200.73; 395/200.58; 370/409

[58] **Field of Search** 395/200.63, 200.58, 395/200.73, 200.62; 370/395, 420, 463, 397, 399, 409

[56] **References Cited**

U.S. PATENT DOCUMENTS

5,175,800	12/1992	Galis et al.	706/45
5,440,551	8/1995	Suzuki	370/395
5,461,611	10/1995	Drake, Jr. et al.	370/420
5,491,742	2/1996	Harper et al.	379/201
5,610,910	3/1997	Focsaneanu et al.	370/351
5,623,488	4/1997	Svennevik et al.	370/360
5,640,399	6/1997	Rostoker et al.	370/466
5,740,075	4/1998	Bigham et al.	395/200.59
5,802,502	9/1998	Gell et al.	705/37
5,828,844	10/1998	Civanlar et al.	395/200.58
5,831,972	11/1998	Chen	370/230

OTHER PUBLICATIONS

Zhang et al., "RSVP: A New Resource ReSerVation Protocol", IEEE Network, Sep. 1993.

Primary Examiner—Frank J. Asta
Assistant Examiner—Daniel Patru

[57] **ABSTRACT**

A method and apparatus for use in a network utilizing Internet Protocol (IP), Resource Reservation Protocol (RSVP), and Asynchronous Transfer Mode (ATM) protocol is provided. An intelligent policy mapping database (PMD) accessible at the network level by both the RSVP and ATM protocol stacks maps RSVP parameters to ATM parameters with input from factors outside of the RSVP or ATM protocol stacks, e.g., general customer data. With the basis of customer data or other information outside of the RSVP and ATM protocol stacks, a network reservation message to the PMD contains RSVP flow specifications which are mapped to correlated ATM Quality of Service (QoS) parameters.

20 Claims, 4 Drawing Sheets

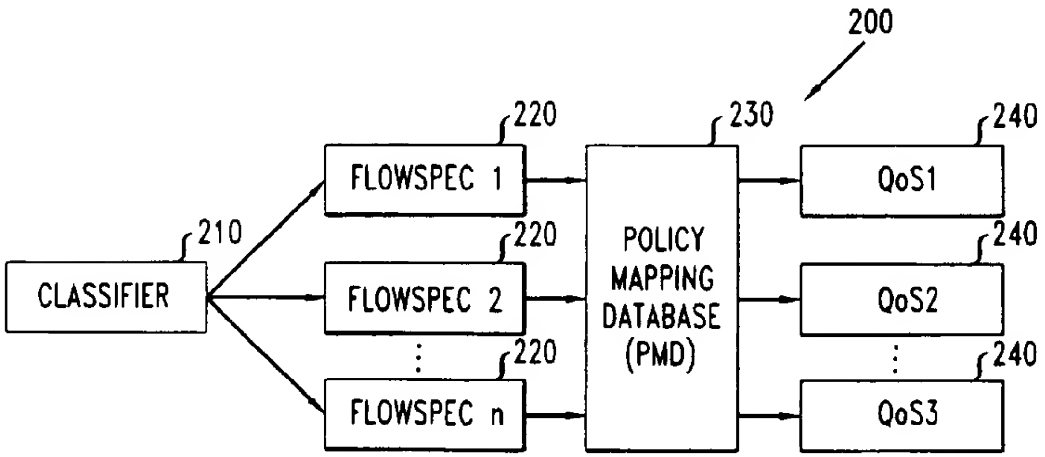


FIG. 1

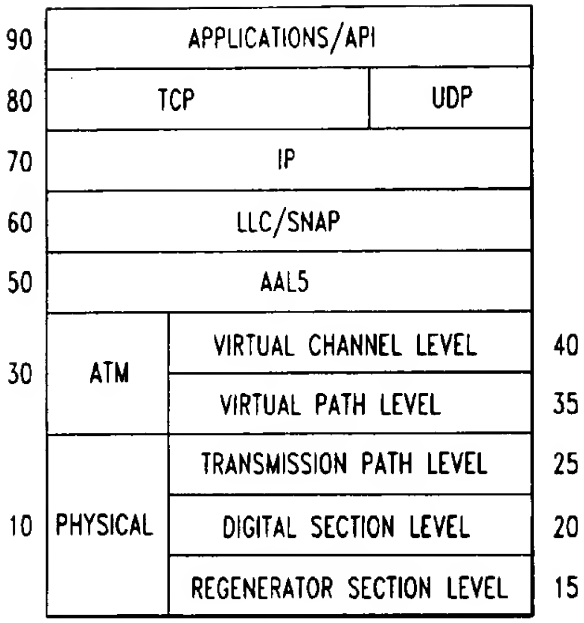


FIG. 2

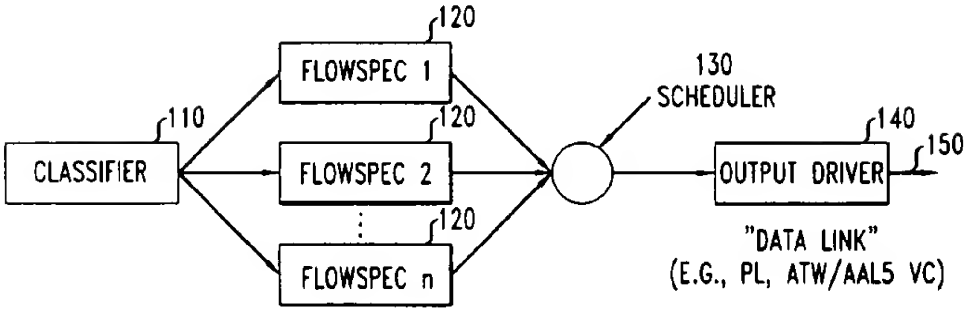


FIG. 3

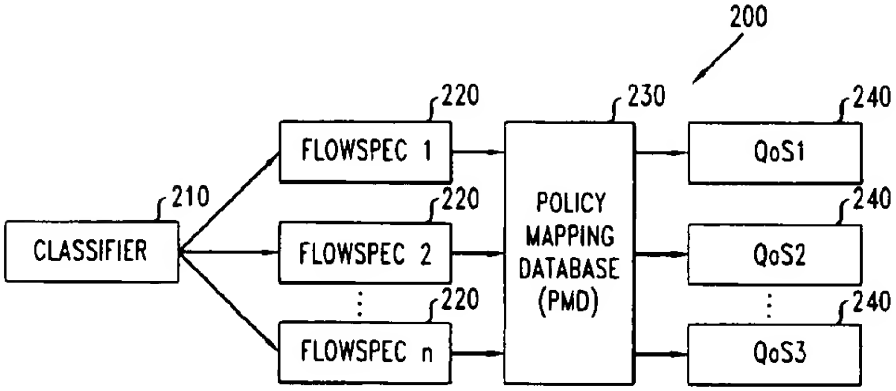


FIG. 4A

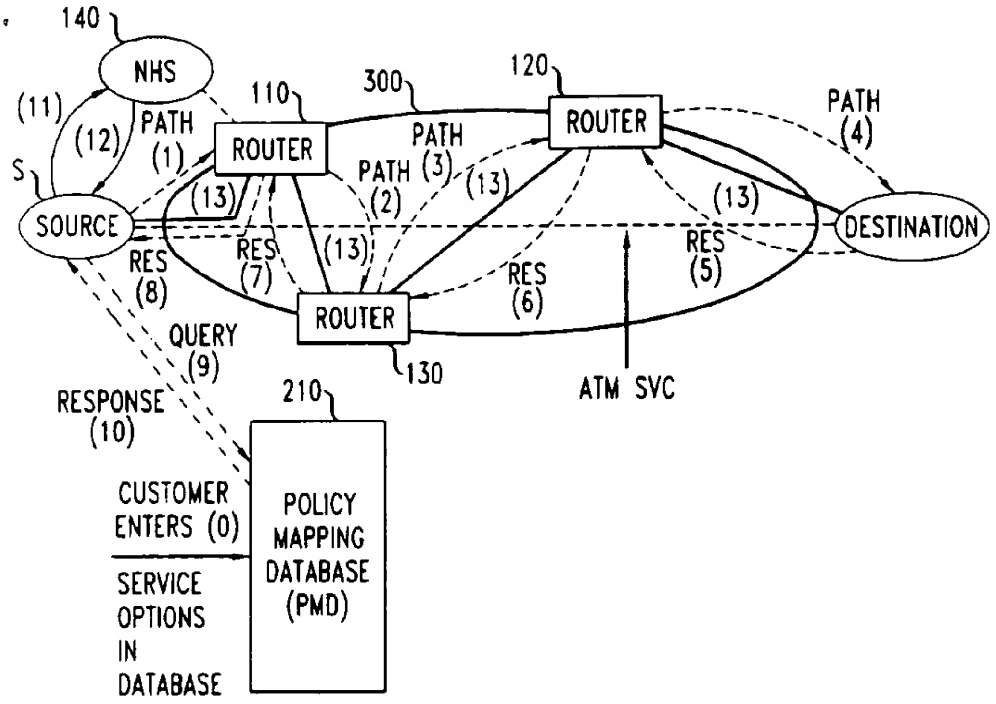

140
NHS
(11)
(12)
PATH
(1)
110
ROUTER
300
PATH
(3)
PATH
(2)
120
ROUTER
PATH
(4)
S
SOURCE
(13)
(13)
(13)
(13)
DESTINATION
RES
(8)
RES
(7)
RES
(6)
RES
(5)
ROUTER
130
QUERY
(9)
RESPONSE
(10)
ATM SVC
210
CUSTOMER
ENTERS (0)
SERVICE
OPTIONS
IN
DATABASE
POLICY
MAPPING
DATABASE
(PMD)

FIG. 4B

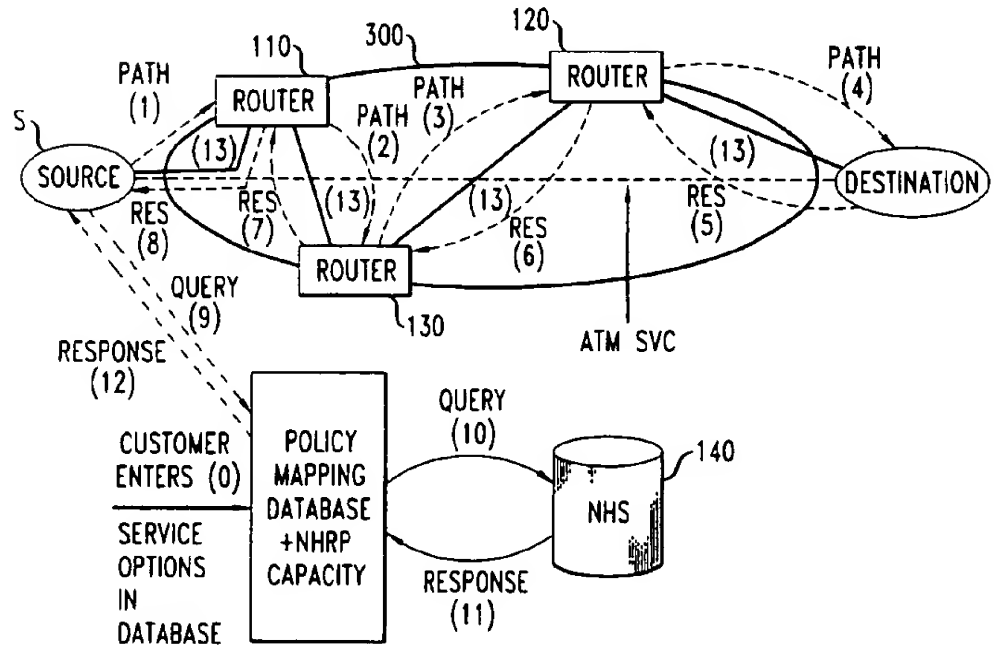

110
300
120
PATH
(1)
ROUTER
PATH
PATH
(3)
ROUTER
PATH
(4)
S
(13)
(2)
(13)
SOURCE
(13)
DESTINATION
RES
(8)
RES
(7)
(13)
(13)
RES
(6)
RES
(5)
ROUTER
130
QUERY
(9)
ATM SVC
RESPONSE
(12)
CUSTOMER
ENTERS (0)
SERVICE
OPTIONS
IN
DATABASE
POLICY
MAPPING
DATABASE
+NHRP
CAPACITY
QUERY
(10)
RESPONSE
(11)
NHS
140

FIG. 4C

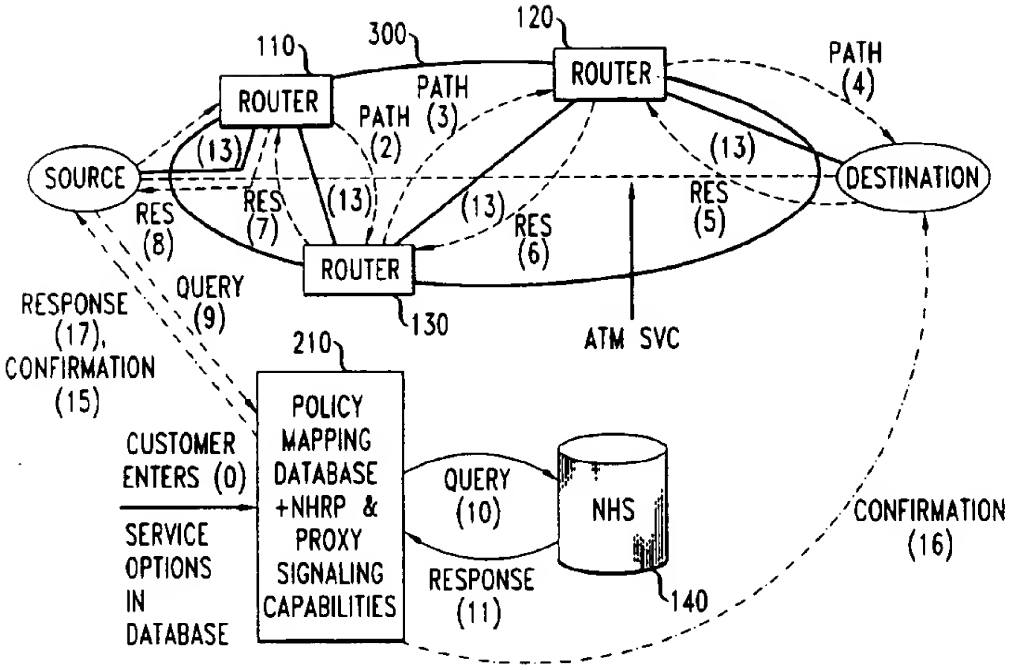

110
300
120
ROUTER
ROUTER
PATH (4)
PATH (3)
PATH (2)
(13)
SOURCE
DESTINATION
RES (8)
RES (7)
(13)
(13)
RES (6)
RES (5)
ROUTER
130
ATM SVC
RESPONSE (17), CONFIRMATION (15)
QUERY (9)
210
POLICY MAPPING DATABASE +NHRP & PROXY SIGNALING CAPABILITIES
CUSTOMER ENTERS (0)
SERVICE OPTIONS IN DATABASE
QUERY (10)
RESPONSE (11)
NHS
140
CONFIRMATION (16)

FIG. 4D

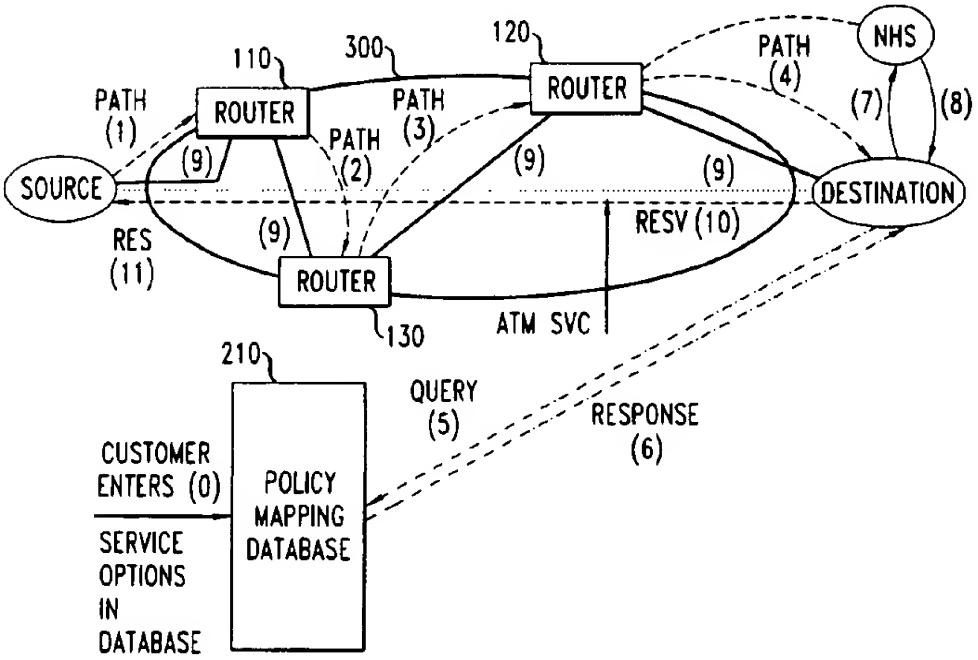

110
300
120
NHS
PATH (4)
PATH (1)
ROUTER
ROUTER
PATH (3)
(7)
(8)
PATH (2)
(9)
(9)
(9)
SOURCE
DESTINATION
RES (11)
(9)
RESV (10)
ROUTER
130
ATM SVC
210
QUERY (5)
RESPONSE (6)
CUSTOMER ENTERS (0)
POLICY MAPPING DATABASE
SERVICE OPTIONS IN DATABASE

FIG. 4E

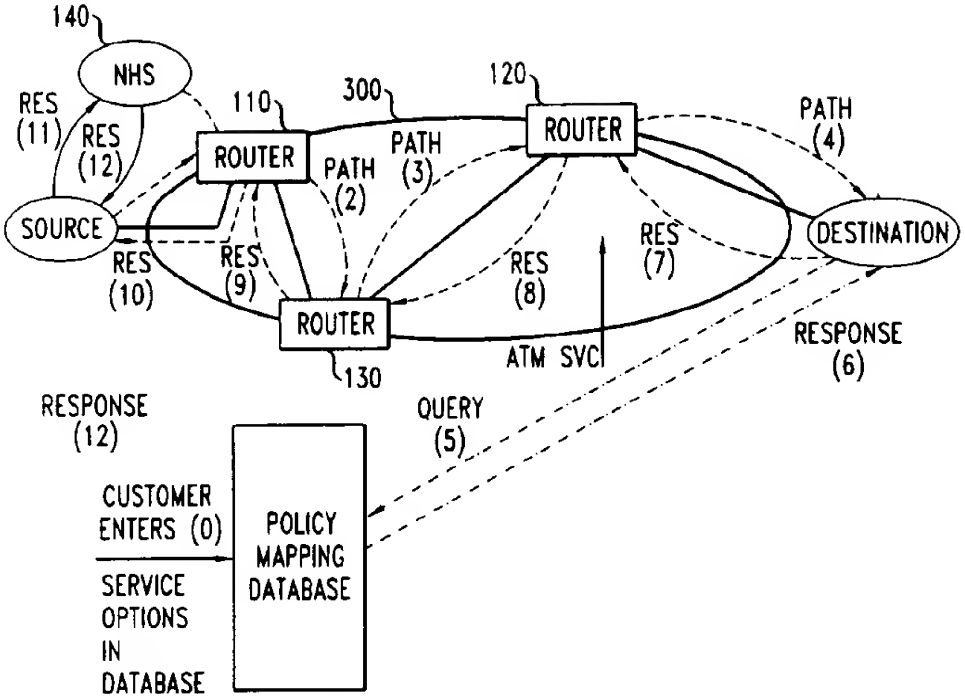


FIG. 5

- ENABLE CUT-THROUGH WITH QoS MAPPING (YES/NO)
- ENABLE CUT-THROUGH WITHOUT QoS MAPPING (YES/NO)
- HOP-BY-HOP QoS MAPPING (NO CUT-THROUGH) (YES/NO)
- NHRP LOOKUP (YES/NO)
  - IF YES
  - 3RD PARTY SETUP (YES/NO)
- RESTRICT CUT-THROUGH (YES/NO)
  - IF YES
  - CUT-THROUGH ONLY FOR DESTINATION IN
    - DOMAIN 1, DOMAIN 2, ...DOMAIN M
    - IPNET 1, IPNET 2, ...IPNET N
- DATE/TOD (TIME OF DAY OVERRIDE)
  - NO CUT-THROUGH
    - FROM T1-T2
    - FROM TN-TN-1
- MULTICAST CUT-THROUGH ALLOWED? (YES/NO)
- BACKUP OPTION (YES/NO)
- USE ALTERNATE ATM PATH WHEN PRIMARY ATM PATH FAILS? (YES/NO)

MANAGEMENT OF ATM VIRTUAL CIRCUITS WITH RESOURCES RESERVATION PROTOCOL

TECHNICAL FIELD

This invention relates to data communications and computer networking.

BACKGROUND OF THE INVENTION

The field of data communications is primarily concerned with how devices talk to each other. As in the case of human beings, for devices to speak to each other they need to use and understand the same language. These languages are called communications protocols. The protocols are agreed upon standards that are normally based on layered models which enable different vendor equipment to communicate (inter-network). A complete exposition of layered model protocols is given in Andrew S. Tannenbaum, Computer Networks, 2nd Edition, Prentice Hall, 1989.

One of the new emerging protocols is the Asynchronous Transfer Mode (ATM) protocol. ATM can be incorporated into one of the most prevalent computer protocols, the Transmission Control Protocol/Internet Protocol (TCP/IP). FIG. 1 displays a conceptual diagram of an IP over ATM Host Protocol Stack. A physical layer **10** of an ATM network consists of a regenerator level **15** which regenerates weakened signals, a digital section level **20** which disassembles and assembles a continuous byte stream, and a transmission path level **25** that assembles and disassembles the payload of the system. Sitting above the physical layer **10** is the ATM layer **30**. The ATM layer is composed of a virtual path level **35** and a virtual channel level **40**. The virtual path level **35** is composed of a bundle of virtual channels that have the same endpoint. The virtual channel level **40** is concerned with issues like the Quality of Service (QoS), the Switched and semi-permanent virtual-channel connections, cell sequence integrity, and traffic parameter negotiation and usage monitoring. The QoS parameter allows ATM to provide network resources based on the different types of applications being used. For example, an application may send out short bursts of information—therefore the application does not require a long connection (e.g., logging on to a remote computer). On the other hand some applications require a large amount of information and not necessarily reliable data transfer (e.g. video conferencing). Utilizing the QoS parameter, in an ATM network a file transfer can be handled differently from a video conference. The AAL5 layer **50** is the segmentation and reassembly sublayer and is responsible for packaging information into cells for transmission and unpacking the information at the other end. The LLC/SNAP layer **60** provides a mechanism for encapsulating other protocols (e.g., IP, Novell IPX) over ATM AAL5. The IP layer **(70)** is the network layer of the IP protocol suite, and provides a common packet format and addressing scheme capable of transporting data over multiple subnetwork technologies (e.g., Ethernet, ATM). The TCP layer **(80)** and UDP (User Datagram Protocol) layer **(80)** provide different types of transport services over IP. Applications **(90)** residing within an end-system may access TCP and UDP services via an applications API (Application Programming Interface).

Communication between devices in a network is performed digitally. The information to be communicated is usually represented as 0's or 1's. The information or data to be communicated (0's and 1's) is usually grouped and separated into units called packets. The layered modeled protocols discussed above are implemented in these packets by defining meaning to bits in the packets, defining different types of packets and defining the sequencing of the different types of packets.

Once data or information has been segmented into packets, the packets are sent into the network where they may take the same path or separate paths. The packets are ultimately recombined at the end device. In ATM, a communications path between two devices is established through a virtual circuit. The circuit is called a virtual circuit because the path may be established and then removed, and resources along the path may be shared by multiple virtual circuits. When the packets are sent through network switches that established virtual circuits through an automated call-setup procedure, the paths are called switched virtual circuits (SVC's).

In an IP packet network, a packet is sent from a transmitting device onto the local network and then transmitted to a device called a router. The router forwards the packet into the network. Current models for IP over ATM have described a method of communicating directly between two end devices (possibly on different subnetworks) once a path between the two devices has been established. An emerging protocol called the Next Hop Resolution Protocol (NHRP) and NHRP servers (NHS's) may be employed to map IP addresses from endpoints on different IP networks to their corresponding ATM addresses. Once the destination ATM address is acquired, a direct ATM path between source and destination may be set up. When the source and destination are members of different subnetworks, such as an ATM SVC is referred to as a cut-through or short cut SVC. Using this approach, all of the packets take the same path (virtual circuit), between the two end devices. However, in an alternative model, the communications connection between two end devices, may be established so that all packets are processed through a router. When packets are processed through a router the packets may not all take the same path. Processing packets through a router is advantageous when the applications being performed are small applications (small in time, small in bandwidth). However, processing becomes difficult when the applications have larger requirements (larger time requirements, larger bandwidth requirements). Therefore when the applications have larger requirements, it is generally advantageous to use a switched virtual circuit between two communicating end devices.

When ATM is used in the TCP/IP environment one of the key issues is SVC connection management. At one end of the spectrum a SVC could be established between all communicating entities. At the other end of the spectrum all communicating entities could be forced to go through a router. Given the diversity of applications each of these solutions would be lacking. Therefore, it would be advantageous to allow SVC management to be controlled by the requirements of the application, specifically the QoS requirement of the application.

In current TCP/IP applications the decision of whether to use an SVC or a router is based on the transmitting and destination IP address. Transport protocols such as TCP and UDP use port numbers to identify an application associated with an IP address. Some port numbers (1–255) are well known and represent services such as host functions, file transfer, and network news. Other port numbers (1024–65535) are not well known and therefore may be used to identify QoS requirements in a communications session.

An alternative mechanism for communicating the QoS requirements of an application is through a currently evolv-

ing Resource reSerVation Protocol (RSVP). RSVP enables the reservation of resources and QoS negotiations in an IP network. RSVP operates within the context of the IP protocol and therefore does not take into account the particular subnetwork technology (e.g., ATM) that IP may be operating over. Hence the resource reservation and QoS negotiations occur between communicating end-systems and network routers. In the RSVP protocol methodology a source would send a path message to a destination address to identify a communications route. The destination would request the reservation of resources for a "flow" along the route. Lastly, if the destinations reservation request is accepted, the flow receives the requested network resources and QoS from the path.

The RSVP methodology is supported by several component parts of the RSVP protocol. The first component part is a flow specification which describes the characteristics of the packet stream sent by the source (e.g., short packets of intermittent frequency for terminal communications, longer packets that are generated at more regular intervals for video teleconferencing). The flow specification specifies the desired QoS and is used to set the packet scheduler parameters. Next, a routing protocol provides communication paths. A setup protocol enables the creation and maintenance of the resources reserved. An admission control algorithm maintains the network load at a proper level by rejecting resource requests that would cause the level to be exceeded. Lastly, a packet scheduler is placed in the routers in the path between the source and destination to assure the right QoS.

FIG. 2 displays a flow model of the RSVP model as it is currently implemented over an IP network. In the current implementation of RSVP, packets are classified based on their "session" and "filterspec" parameters (based on among other things, source and destination addresses), and service by the IP protocol is based on the flow specification (referred to as a "flow" for short). The flow model of FIG. 2 shows packet processing within a router. A classifier **110** separates communicating packets based on their "session" and "filterspec" parameters as shown by **120**. The packets are then channeled into a packet scheduler **130** for processing by an output driver **140**, which outputs the data at **150**, the interface leading to the "next-hop" router in the path to the destination (or the destination itself in the case when the next-hop is the destination).

SUMMARY OF THE INVENTION

The present invention defines a method and architecture for implementing RSVP over IP/ATM. In the proposed architecture the resources necessary for a communications pathway between two devices are defined by applications resource requirements. The applications requirements are mapped into RSVP parameters via an RSVP and IP capable Application Programming Interface (API), resident in the host computer. When ATM is used in a network, a straightforward approach is to translate RSVP flow specification parameters to corresponding ATM QoS parameters (Note that the mapping of RSVP parameters to ATM parameters do not correspond exactly). Thus if X represents the set of RSVP parameters, and Y is the set of ATM QoS parameters, then Y=F(X), where F is a function that maps X onto Y.

The present invention augments the mapping of RSVP to ATM by utilizing a database (D) as an interface between the RSVP parameters and the ATM parameters. The database (D) contains user end point (also referred to as a customer) data, used to characterize the network requirements of the customers. Therefore the mapping of RSVP parameters to ATM QoS parameters now takes the form Y=F(X, D), where X, Y and F have the same meaning as above, and D is a policy mapping database (PMD). This database is consulted whenever a mapping from X to Y is to be performed. The use of the database permits decisions and choices to be made that are not possible when a straight-forward mapping from X to Y is employed. For example, the policy mapping database (PMD) gives a user the ability to enable a "short-cut" SVC with QoS mapping or without QoS mapping, disable a "short-cut" SVC establishment and support hop-by-hop QoS mapping. The database enables the user to define whether the Next Hop Resolution Protocol(NHRP) is invoked or not; whether a SVC backup should be established, a Time-of-day override should be implemented, or an alternate ATM path should be used when the primary ATM path fails.

In addition, the mapping from X to Y can depend on customers subscription to different levels of security, priority and performance, time of day information, and information about the network state. In summary RSVP flow specifications are mapped to ATM switched virtual circuits with specified QoS utilizing a policy mapping database (PMD). As a result of mapping the flow specifications of RSVP to the QoS requirements of ATM, the proposed method and architecture enables the implementation of RSVP over IP over ATM.

BRIEF DESCRIPTION OF THE DRAWINGS

The objects, advantages and novel features of the invention will be more fully apparent from the following detailed description when read in connection with the accompanying drawings wherein:

FIG. 1 displays the TCP/IP protocol stack over the Asynchronous Transfer Mode protocol stack.

FIG. 2 displays RSVP operation over a non-QoS capable sub-network.

FIG. 3 displays a flow model of the RSVP protocol over IP over ATM with a Policy Mapping Database (PMD) used to manage the protocol translation.

FIG. 4A describes a detailed communication between a source, a destination, the PMD, and an Address resolution/Next Hop server (NHS), utilizing the methodology of the present invention, where the source queries the NHS.

FIG. 4B describes a detailed communication between a source, a destination, the PMD, and an Address resolution/Next Hop server (NHS), utilizing the methodology of the present invention, where the PMD queries the NHS.

FIG. 4C describes a detailed communication between a source, a destination, the PMD, and an Address resolution/Next Hop server (NHS), utilizing the methodology of the present invention, where the PMD queries the NHS and sets up the connection between the source and destination using third party call setup mechanisms.

FIG. 4D shows a scenario where the destination station queries the PMD and sets up an SVC.

FIG. 4E details a scenario where the destination queries the PMD and No Cut-Through SVC results.

FIG. 5 details the contents of the PMD.

DETAILED DESCRIPTION OF THE INVENTION

The present invention is directed to a method and architecture for allocating network resources (e.g., bandwidth, priority) based on the type of application that is being used

at the communicating endpoints. The Asynchronous Transfer Mode (ATM) Architecture and the RSVP protocol in combination have the necessary components to enable the allocation of network resources based on the application. In the ATM protocol traffic descriptors and the Quality of Service (QoS) feature can be used to establish different network requirements based on the application. For example, since a telnet session uses smaller infrequent packets which could be transferred through a router, a set of traffic descriptors and Quality of Service parameters that defines this kind of connection could be established. However, if a video conference is established, large, delay sensitive packets would be frequently generated. Therefore, a dedicated switched virtual circuit with low delay sensitive characteristics would be more efficient to carry on a video conferencing session.

The RSVP protocol is implemented with components that complete the requirements necessary to create bandwidth allocations based on QoS. The RSVP protocol includes classifiers, which classify the packets, and flow specifications which define and detail relevant characteristics of the packets. Lastly, the RSVP parameters and flow specifications are mapped to ATM switched virtual circuits with corresponding traffic descriptors and QoS parameters using a policy mapping database(PMD). The policy mapping database correlates the RSVP flow specifications to the ATM QoS parameters of switched virtual circuits.

FIG. **3** displays a flow model of the present invention. The flow model of RSVP over ATM correlates the flow specifications and QoS Switched virtual circuits, to establish ATM SVC's. In FIG. **3** a classifier **210** classifies packets based on their session and filterspec parameters. Each classification of packets has an associated flow specification **220**. The flow specifications **220** are fed into a policy mapping database **230**. The policy mapping database (PMD) **230** maps the packets based on the flow specification **220**. The PMD defined by **230** enables the mapping between the RSVP flow specification parameters **220** and the ATM QoS parameters **240**. The mapping of flow specifications **220** to ATM SVC's, **240** is based on the resources required by the application (e.g., best effort traffic may be mapped to a router, and video conferencing would be mapped to separate SVC with appropriate traffic descriptors and QoS parameters). A user interface to the policy mapping database (PMD) can be established to allow users to manage the mapping for their own traffic.

FIG. **4**A gives an end-to-end flow diagram of the sequence of steps for the unicast (transmission of a packet from one end point to another endpoint) case. In the disclosed methodology the customers initially enters the various service options in the policy mapping database (0), together with the list of IP end-points to which the options apply (Note that a single customer may have multiple entries in the database, i.e., one subnet of the customer end-points may have different options than another subnet). When source (S) wishes to communicate with destination (D), the source (S) sends a path message (1) directed towards (D) via its next-hop router **110**. The path message (1) is then forwarded hop-by-hop towards D via steps (2), (3), and (4). After D receives the path message it returns a reservation request back to S hop-by-hop in the reverse direction using the route taken by the path message in steps (5), (6), (7) and (8). The route information necessary for these steps is maintained in routers **110**, 120, and **130** as PATH state information. When S, receives the reservation, it sends a query (9) to the PMD denoted by **210**. The PMD does not have to be physically co-located with S. The PMD is reachable via its ATM address, which is known to S. The query (9) contains all the information from the original received reservation messages. Query (9) is processed by the PMD **210** and a response (10) containing the required ATM traffic descriptors and QoS parameters, and information pertaining to the various PMD specified options is returned to S. The details of the query and response messages from S to the PMD are specified below. Once S has the results of the response (10), and assuming the service options, network state, etc., permit cut-through, S sends an NHRP query (11) to its default NHS **140** and receives a response (12) containing D's ATM address. S then sets-up an ATM SVC (13) directly to D using the ATM QoS information from response (10).

The methodology and architecture disclosed in FIG. **4**A can be modified to improve performance and conduct processing on behalf of end-system clients. In the architecture of FIG. **4**B, operation is identical to that of FIG. **4**A up to and including query message (9). After receiving query (9), the PMD initiates an NHRP request (10) to the NHS on behalf of the source. The NHRP request (10) is signaled by an NHRP lookup option in the PMD query message (9), which also contains the IP address of D. When the PMD receives NHRP reply (11) for the NHS, it responds with (12) to the source. The response now additionally contains the ATM address corresponding to the IP address of D. The source S is now able to setup a call (13) directly to destination D, without having to go through the step of consulting an NHRP server, because the NHRP server was accessed by the PMD **210**, or may be located in the PMD **210**.

A further efficiency is possible when an NHRP lookup option is enabled. As shown in FIG. **4**C, a 3rd party ATM call setup is initiated by the PMD **210** via proxy signaling (i.e., when a party other than the communicating endpoints signals the endpoints for communication established), denoted by (12). Proxy signaling is signaled by the source S to the PMD **210** by a 3rd party/proxy signaling call setup option in the PMD query message (9) (it is assumed that the PMD has been provisioned to carry-out proxy signaling on behalf of both S and D; this requires that a "signaling" Virtual Circuit be set-up between the PMD **210** and the communicating endpoints). In FIG. **4**C, operation is identical to that of FIG. **4**B, up to and including reply (11). Afterward the PMD **210** initiates a 3rd party ATM call setup request (12) via proxy signaling to the ATM network on behalf of both S and D. A connection (13) is then setup between S and D. When the connection setup is complete, a proxy signaling confirmation message (14) is received from the ATM Network **300**. The PMD **210** then issues proxy signaling confirmation messages (15) to (D) and (16) to (S). The confirmation messages (15), (16) may include Virtual Path/Virtual Channel Identifier (VPI/VCI), addressing information, QoS, and other information received in message (14). The confirmation (15) to the source S may be piggy-backed in the PMD response (17), so that a separate message need not be sent. The source S is now able to send to destination D using the VPI/VCI information received in message (14) without either having to do an IP to ATM address translation, or an ATM call setup request.

A further modification to the methodology disclosed above results in a significant improvement to the overall performance of the system. The improvement results from not reserving resources in the intermediate routers (**110**, **120**, **130**) between the source S and the destination D in case a cut-through is permitted and an SVC is established between S and D. To explain this point, observe that in the above scenarios the RSVP Reservation Request message that was returned back from D to S results in the reserving of

resources at each router (**110, 120, 130**) along the path from D to S. In case the end result is to permit cut-through and establish an ATM SVC between S and D, two issues result. First a mechanism should be used to free any reserved resources along the path defined through routers **120, 130,** and **140**. This can be simply achieved either through a time-out mechanism or by letting S send an RSVP Reservation Teardown message. The second more significant issue is that new reservations that actually require resources in these intermediate routers (e.g., because cut-through is not permitted for these reservations) may be blocked because of lack of resources in routers **110, 120,** and **130,** or associated links in the network, This issue is addressed by allowing D to query the PMD. If cut-through is permitted, then D establishes the SVC to S and sends its RSVP Reservation Request message to S over the SVC. In other words, no RSVP Reservation Request message is sent back hop-by-hop in the reverse direction along the route taken by the RSVP Path message, and no resources are reserved in the intermediate routers. Note that clearing the path state information in the intermediate routers (that was created when processing the Path message) is still required. This is not considered a problem because no considerable amount of memory is required to store the path state information.

There are still two issues to resolve at S, when using D to set-up the SVC. The first issue is how to associate the received RSVP Reservation Request message to the Path message that was previously sent. Observe that if D has queried the PMD and cut-through is permitted, the RSVP Reservation Request message is received by S over a virtual circuit which is different from the virtual circuit over which the RSVP Path message was sent. The RSVP protocol associates RSVP PATH and RSVP Reservation messages by using a message ID field in these messages. We additionally require the use of unique message ID over a single interface. The same message ID should not be used over separate virtual circuits supported on the interface to identify different reservations.

The second issue arises when S receives an RSVP Reservation Request message over the same virtual circuit that was used to send the RSVP Path message. The problem S faces is how to distinguish between the following two cases. The first case is that a query to the PMD was performed by D and cut-through was not permitted. In this case, a second query to the PMD by S must be avoided. The second case is that a query to the PMD was not performed by D and a query to the PMD by S must be performed. We solve this problem by defining an end-to-end user signaling mechanism between S and D. The method disclosed in the present invention advocates the use of an RSVP Object with an unassigned Class-Num (e.g., 64<Class-Num<128 are unassigned) to signal whether or not a query to the PMD was performed. Consistent with the RSVP specification, systems that do not recognize the Object will quietly ignore it. However systems that implement the methodology that is disclosed in the instant invention will recognize the Object and act on its information. In other words, the disclosed methodology advantageously utilizes unused bits in the RSVP protocol packet structure as a signaling mechanism to communicate information between sources and destinations.

An example of the methodology defined above is presented in FIG. **4***d* and FIG. **4**E. In FIG. **4**D, operation is identical to that of FIG. **4**A up to and including Path message (**4**). D sends a query (**5**) to the PMD **210**. The query (**5**) contains the information from the Path message and D's requested reservation. Query (**5**) is processed by the PMD **210** and a response (**6**) containing the required ATM traffic descriptors, QoS parameters and the various options is returned to D. Once D has the results of the response (**6**) and cut-through is permitted (scenario when cut-through is not permitted is discussed below), it sends an NHRP query (**7**) to its default NHS and receives a response (**8**) containing the ATM address of S. D then sets up an ATM SVC (**9**) directly to S using the ATM QoS information from response (**6**). D then sends the RSVP Reservation Request message (**10**) to S over the established ATM SVC. S associates the RSVP Reservation Request message (**10**) to the RSVP Path message (**1**) by the use of the message ID fields in messages (**1**) and (**10**).

In FIG. **4**E, operation is identical to that of FIG. **4**D up to and including the query response (**6**) returned by the PMD **210** to D. Once D has the results of the response (**6**) and cut-through is not permitted, D sends an RSVP Reservation Request message via Res (**7**), Res (**8**), Res (**9**) and Res (**10**) to S, using the stored path state information in routers **120, 130,** and **110** along the path from D to S. When S receives the reservation (**10**), it determines that a query to the PMD **210** was performed (communicated via the Object with 64<Class-Num<128 mechanism described above) and that no query to the PMD **210** by S is needed. However if S determines that a query to the PMD **210** was not performed at D, it sends a query to the PMD **210**. This is the case illustrated in FIG. **4**A.

Note that the methodology described in FIGS. **4**A to **4**E applies to the IP unicast case (i.e., a single transmitter sending IP packets to a single receiver). The methodology described in FIGS., **4**D and **4**E (where the destination rather than the source queries the PMD) **210** is also appropriate for the IP multicast case (i.e., a single transmitter sends IP packets to a group of receivers). In the multicast case, receivers decide whether or not to join a particular multicast group. The transmitter may not even be aware of which receivers are receiving its transmission. The receiver-driven nature of IP multicast is therefore well accommodated by having receivers/destinations query the PMD **210,** as shown in FIGS. **4***d* and **4***e*. When cut-through is permitted, it is accomplished by a ATM level leaf-initiated join operation to either the source or an intermediate multicast server. Note further that just as in the unicast case, in the multicast case the PMD **210** may both resolve IP addresses to ATM addresses and perform third party multipoint call setup functions on behalf on the querying entity. Furthermore if PMD **210** performs such functions it may also implement address screening thereby providing an optional security function to multicast (and unicast) communication.

FIG. **5** details the contents of the Policy Mapping Database (PMD). For each set of IP end-points, a customer enters whether or not to:

(i) Enable cut-through with QoS mapping: When enabled, ATM cut through to the destination is always attempted. There is a mapping of RSVP flow specification parameters to ATM QoS parameters and traffic descriptors. Depending on the reservation style of RSVP, the mapping of RSVP flows to ATM. VCs may be 1 to 1 or many to 1.

(ii) Enable cut-through without QoS mapping: When enabled, ATM cut-through to the destination is always attempted. The ATM SVC is always "best-effort" with no QoS parameters or traffic descriptors being set. The mapping is typically many to 1.

(iii) Disable cut-through, support hop-by-hop QoS mapping: When enabled, ATM cut-through is not attempted. Packets are forwarded hop-by-hop accord-

ing to the Classical IP router-based packet forwarding model. The mapping of RSVP flow specification parameters to ATM QoS parameters and traffic descriptors takes place on a hop-by-hop basis. Depending on the reservation style of RSVP, the mapping of RSVP flows to ATM VCs may be 1 to 1 or many to 1.

(iv) NHRP Lookup Option

If No, then ignored.

If Yes, then the PMD invokes the NHRP server (NHS) on behalf of the source and returns the result of the query (i.e., the ATM address of the destination) back to the source in the PMD response message.

Third party setup by proxy signaling option

If No, then ignored.

If Yes, then the PMD, having invoked the NHRP server (NHS) on behalf of the source, also sets up an ATM connection between the Source and Destination using proxy signaling. If enabled, this sub-option assumes that both the Source and Destination have been provisioned to allow the PMD to act as a proxy signaling agent for each.

(v) Restrict cut-through option:

If No, then ignored.

If Yes, then the customer lists the set of destination domain names, and IP addresses for which cut-through is allowed for the given set of IP end-points. Address prefixes and domain names suffices are permitted. Cut-through is not attempted to destinations not in this list.

(vi) Day/Time-of-day override:

If No, then ignored.

If Yes, then customer enters the days of the month and times of the day, for which cut-through is not to be attempted.

(vii) Multicast cut-through allowed:

If No, then cut-through is not attempted for multicast destinations (i.e. if the destination address is a class D address).

If Yes, then cut-through is permitted. The customer may also list the set of class D addresses for which cut-through is allowed, and the list of ATM end-points that may join a particular multicast group.

(viii) Backup Option:

If Yes, then

(a) if current set of options require cut-through, and if cut-through fails, attempt a hop-by-hop setup.

(b) if current set of options require hop-by-hop and RSVP setup fails, attempt a cut-through.

(ix) Use alternate ATM path when primary ATM path fails:

If No, then ignored.

If Yes, then assuming that the destination is reachable from the source by 2 or more distinct ATM networks, and that PNNI routing between these networks is not employed, the source may have 2 or more distinct ATM addresses for a given IP destination. When a call setup attempt to the first ATM address fails, a second, third, etc., attempt is made to the second, third, etc., ATM address, until an attempt succeeds, or there are no more alternate addresses.

The above options may be grouped together to provide different levels/categories of service to customers. For example, one (high) level of service might consist of enabling options (i), (vii), and (viii), while another level of service might consist of enabling options (ii), (iv), and (v). Further the details of each option may change, and additional options may be added to the database without changing the overall operation of the invention.

An exemplary database query and response message contains the following parameters:

Query

1. The RSVP flow specification including both the traffic specifications (amount and characteristics of bandwidth needed), service specific parameters (e.g., packet delay, packet jitter, packet loss), and the filter specification identifying and characterizing the stream of packets in the packet flow.

Response

1. The contents of the query as described above.
2. ATM related parameters
   Cut-through or not
   ATM traffic descriptors
   ATM QoS parameters
   Backup enabled (Yes/No)
   Alternate ATM (Yes/No)
   If NHRP lookup is enabled, the ATM address corresponding to the target IP address of the Query message.
   If 3rd party setup is enabled, then the ATM Virtual Path/Virtual channel identifiers to use (VPI/VCI) to reach the IP target.

Note that the query and response message content can be extended to allow override of some of the provisioned entries in the PMD on a call-by-call basis. For example one could request NHRP lookup for some connections but not for others. The query messages are simply augmented by the addition of the various options. It should also be appreciated that the system can also operate successfully even if the contents of the query message contains a subset (e.g. port number and destination address) of information. For example, if information pertaining only to the filter-specification was available it is still possible to accomplish a mapping from destination address and port number to ATM QoS. This enables the system to be used even when RSVP is not employed.

While several embodiments of the invention are disclosed and described, it should be appreciated that various modifications may be made without departing from the spirit of the invention or the scope of the subjoined claims.

We claim:

1. A method of operating a network capable of utilizing Internet protocol, resource reservation protocol, and asynchronous transfer mode protocol, based on auxiliary user information, said method comprising:

providing a policy mapping database with said auxiliary user information; and

mapping parameters in said resource reservation protocol to parameters in said asynchronous transfer mode protocol utilizing said auxiliary information in said policy mapping database;

wherein said auxiliary user information is a set of policies, definable by an end user, regarding a pathway of communication of packets sent or received by said end user over said network.

2. The method as claimed in claim 1, wherein said mapping comprises:

classifying packets in said network thereby creating classified packets;

separating said classified packets based on said parameters in said resource reservation protocol; and

correlating said parameters in said resource reservation protocol with said parameters in said asynchronous

transfer mode protocol utilizing said auxiliary information in said policy mapping database.

3. The method as claimed in claim 1, wherein:

said parameters in said resource reservation protocol are flow specifications.

4. The method as claimed in claim 1, wherein:

said parameters in said asynchronous transfer mode protocol are at least one of quality of service parameters and traffic descriptors.

5. A method of operating a network based on information regarding user applications, said method comprising:

defining a communications path, for use by a user application, with a path message;

utilizing a reservation message to reserve resources along said communications path, said reservation message containing flow specifications;

mapping said flow specifications to related parameters in an asynchronous transfer mode protocol utilizing a policy mapping database containing auxiliary information regarding said user application, said policy mapping database being accessible from said network; and

establishing a switched virtual circuit based on said mapping;

wherein said auxiliary user information is a set of policies, definable by an end user, regarding a pathway of communication of packets sent or received by said end user over said network.

6. The method as claimed in claim 5, wherein:

said auxiliary information is information customized on a user by user level.

7. The method as claimed in claim 5, wherein:

said policy mapping database is centralized for access from a network level.

8. The method as claimed in claim 5, wherein:

said flow specifications include at least one of traffic specification service specific parameters, and a filter specification which identifies and characterizes a stream of packets.

9. The method as claimed in claim 5, wherein:

said related parameters in said asynchronous transfer mode protocol include at least one of asynchronous transfer mode protocol traffic descriptors and asynchronous transfer mode protocol quality of service parameters.

10. The method as claimed in claim 5, wherein said auxiliary information regarding said user application includes at least one of:

cut-through enable with quality of service mapping;

cut-through enable without quality of service mapping;

cut-through disable;

next hop resolution protocol lookup;

third party setup;

restricted cut through;

multicast cut-through enabled;

allowable multicast group and group membership;

backup enabled; and

alternate asynchronous transfer mode protocol path enabled.

11. A method of operating a network based on information regarding user applications, resources in said network including a source, a destination, a next hop resolution protocol server, and a policy mapping database including auxiliary user information regarding a plurality of users, said method comprising:

sending a path message from said source to said destination;

returning a reservation message, based on said path message, from said destination to said source;

sending a query message from said source to said policy mapping database, said query message containing resource reservation protocol information acquired from at least one of said path message and said reservation message;

returning a response message, based on said query message, from said policy mapping database to said source, said response message containing asynchronous transfer mode protocol parameters determined utilizing said auxiliary user information in said policy mapping database;

utilizing said response message to generate a query request from said source to said next hop resolution protocol server to determine an asynchronous transfer mode protocol address for said destination,

transmitting an asynchronous transfer mode protocol address from said next hop resolution protocol server to said source; and

utilizing said asynchronous transfer mode protocol address to establish a switched virtual circuit from said source to said destination;

wherein said auxiliary user information is a set of policies, definable by an end user, regarding a pathway of communication of packets sent or received by said end user over said network.

12. A method of providing network resources based on information regarding user applications, said network resources including a source, a destination, a next hop resolution protocol server, and a policy mapping database including auxiliary user information regarding a plurality of users, said method comprising:

transmitting a path message from said source to said destination;

returning a reservation message from said destination to said source;

transmitting a query message from said source to said policy mapping database, said query message containing resource reservation protocol information acquired from at least one of said path message and said reservation message;

transmitting a query message from said policy mapping database to said next hop resolution protocol server;

returning an asynchronous transfer mode protocol address for said destination to said policy mapping database;

returning a response message from said policy mapping database to said source, said response message containing at least one of asynchronous transfer mode protocol parameters and said asynchronous transfer mode protocol address of said destination determined utilizing said auxiliary user information in said policy mapping database; and

establishing a switched virtual circuit between said source and said destination based on said response message;

wherein said auxiliary user information is a set of policies, definable by an end user, regarding a pathway of communication of packets sent or received by said end user.

13. A method of providing network resources based on information regarding user applications, said network resources including a source, a destination, a next hop

resolution protocol server, and a policy mapping database containing auxiliary user information regarding a plurality of users, said method comprising:

transmitting a path message from said source to said destination;

returning a reservation message from said destination to said source;

transmitting a query message from said source to said policy mapping database, said query message containing resource reservation protocol information acquired from at least one of said path message and said reservation message;

transmitting a query message from said policy mapping database to a next hop resolution protocol server, said query message including information obtained from said auxiliary user information in said policy mapping database;

returning an asynchronous transfer mode protocol address for said destination to said policy mapping database; and

utilizing said policy mapping database as a third party proxy to establish a switched virtual circuit connection between said source and said destination;

wherein said auxiliary user information is a set of policies, definable by an end user, regarding a pathway of communication of packets sent or received by said end user.

14. A method of providing network resources based on information regarding user applications, said network resources including a source, a destination, a resource reservation protocol server, and a policy mapping database containing auxiliary user information regarding a plurality of users, said method comprising:

transmitting a path message from said source to said destination;

transmitting a query message from said destination to said policy mapping database, said query message containing resource reservation protocol information acquired from at least one of said path message and from said destination;

returning a response message from said policy mapping database to said destination, said response message including at least one of asynchronous transfer mode protocol traffic descriptors and quality of service parameters determined at least in part based on auxiliary user information contained in said policy mapping database;

utilizing said destination to query said next hop resolution protocol server for an asynchronous transfer mode protocol address of said source;

establishing a switched virtual circuit between said destination and said source utilizing at least one of said asynchronous transfer mode protocol address of said source and said response message; and

returning a reservation message from said destination to said source via said switched virtual circuit;

wherein said auxiliary user information is a set of policies, definable by an end user, regarding a pathway of communication of packets sent or received by said end user.

15. A method of providing network resources based on information regarding user applications, said network resources including a source, a destination, a next hop resolution protocol server, and a policy mapping database containing auxiliary user information, said method comprising:

transmitting a path message from said source to said destination;

transmitting a query message from said destination to said policy mapping database, said query message containing resource reservation protocol information acquired from at least one of said path message and said destination;

transmitting a query message from said policy mapping database to a next hop reservation protocol server on behalf of said destination with a sender IP address as a target;

returning an asynchronous transfer mode protocol address for said source to said policy mapping database;

returning a response message from said policy mapping database to said destination, said response message including at least one of resource reservation protocol parameters, asynchronous transfer mode protocol parameters and said asynchronous transfer mode protocol address of said source, determined at least in part based on said auxiliary user information contained in said policy mapping database;

establishing a switched virtual circuit between said destination and said source utilizing at least one of said asynchronous transfer mode protocol address of said source, said asynchronous transfer mode protocol parameters and said resource reservation protocol parameters; and

returning a reservation message from said destination to said source via said switched virtual circuit;

wherein said auxiliary user information is a set of policies, definable by an end user, regarding a pathway of communication of packets sent or received by said end user.

16. A method of providing network resources based on information regarding user applications, said network resources including a source, a destination, a next hop resolution protocol server, and a policy mapping database containing auxiliary user information, said method comprising:

transmitting a path message from said source to said destination;

transmitting a network level query message from said destination to said policy mapping database, said query message containing resource reservation protocol information acquired from at least one of said path message and from said destination;

transmitting a query message from said policy mapping database to a next hop resolution protocol server on behalf of said destination with a sender IP address as a target;

returning an asynchronous transfer mode protocol address for said source to said policy mapping database;

utilizing said policy mapping database containing auxiliary user information as a third party proxy to set-up a switched virtual circuit connection between said source and said destination;

returning a response message from said policy mapping database to said destination, said response message including at least one of resource reservation protocol parameters, asynchronous transfer mode protocol parameters, said asynchronous transfer mode protocol address, and a virtual path identifier/virtual channel identifier of said source; and

returning a reservation message from said destination to said source via said switched virtual circuit;

wherein said auxiliary user information is a set of policies, definable by an end user, regarding a pathway of communication of packets sent or received by said end user.

17. An Internet protocol over asynchronous transfer mode protocol network comprising:

a plurality of Internet protocol packet classifiers, each packet classifier having an assigned flow specification;

a plurality of quality of service based switched virtual circuits; and

a policy mapping database relating said plurality of Internet protocol packet classifiers to respective ones of said plurality of quality of service based switched virtual circuits based on auxiliary user information contained in said policy mapping database;

wherein said auxiliary user information is a set of policies, definable by an end user, regarding a pathway of communication of packets sent or received by said end user.

18. The method of providing network resources based on information regarding user applications according to claim 14, further comprising:

utilizing unused bits in said reservation message to signal to said source that a cut-through has already been performed.

19. The method of providing network resources based on information regarding user applications according to claim 15, further comprising:

utilizing unused bits in said reservation message to signal said source that a cut-through has already been performed.

20. The method of providing network resources based on information regarding user applications according to claim 16, further comprising:

utilizing unused bits in said reservation message to signal said source that a cut-through has already been performed.

* * * * *

US006084879A

United States Patent [19]

Berl et al.

[11] Patent Number: 6,084,879

[45] Date f Patent: *Jul. 4, 2000

[54] **TECHNIQUE FOR CAPTURING INFORMATION NEEDED TO IMPLEMENT TRANSMISSION PRIORITY ROUTING AMONG HETEROGENEOUS NODES OF A COMPUTER NETWORK**

[75] Inventors: **Steven H. Berl**, Piedmont; **Ulrica Tam**, Belmont, both of Calif.

[73] Assignee: **Cisco Technology, Inc.**, San Jose, Calif.

[*] Notice: This patent is subject to a terminal disclaimer.

[21] Appl. No.: **09/354,360**

[22] Filed: **Jul. 14, 1999**

Related U.S. Application Data

[63] Continuation of application No. 08/833,834, Apr. 10, 1997, Pat. No. 5,940,390.

[51] **Int. Cl.**[7] **H04L 12/56**

[52] **U.S. Cl.** **370/389**; 370/389; 370/469; 370/401

[58] **Field of Search** 370/236, 384, 370/389, 400, 401, 465, 469, 466, 522, 410, 412, 414

[56] **References Cited**

U.S. PATENT DOCUMENTS

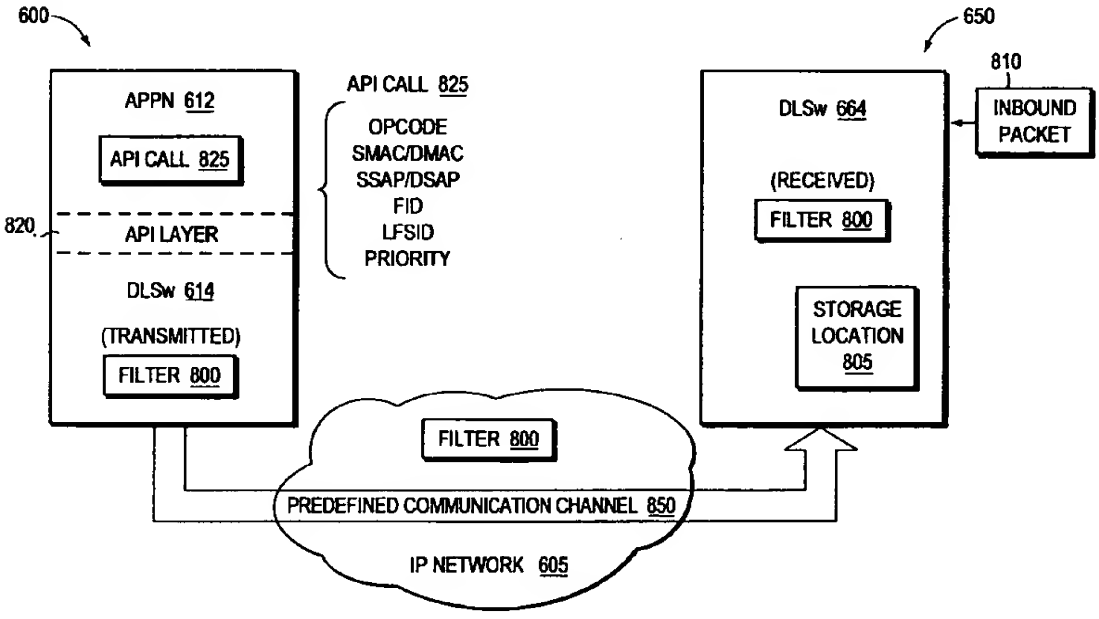
4,484,326	11/1984	Turner	370/60
4,775,973	10/1988	Tomberlin et al.	370/60
5,218,676	6/1993	Ben-Ayed et al.	395/200
5,251,209	10/1993	Jurkevich et al.	370/82
5,317,568	5/1994	Bixby et al.	370/85.6
5,319,641	6/1994	Fridrich et al.	370/451
5,416,769	5/1995	Karol	370/414
5,517,620	5/1996	Hashimoto et al.	395/200.15
5,541,922	7/1996	Pyhalammi et al.	370/404
5,664,105	9/1997	Keisling et al.	395/200.54
5,671,224	9/1997	Pyhalammi et al.	370/401
5,737,526	4/1998	Periasamy et al.	395/200.06
5,748,925	5/1998	Waclawasky et al.	395/311
5,768,271	6/1998	Seid et al.	370/389
5,781,726	7/1998	Pereira	395/200.3
5,848,233	12/1998	Radia et al.	395/187.01
5,892,924	4/1999	Lyon et al.	395/200.75
5,940,390	8/1999	Berl et al.	370/389

Primary Examiner—Chi H. Pham
Assistant Examiner—Maikhanh Tran
Attorney, Agent, or Firm—Cesari & McKenna, LLP

[57] **ABSTRACT**

A mechanism conveys information pertaining to transmission priority (TP) levels of inbound packets transmitted over a heterogeneous network from a switching node to a hybrid node of the network. The mechanism comprises a packet-recognizing filter having a format that is generated by the hybrid node and dynamically transmitted to the switching node over a predefined communication channel of the network. The filter enables the switching node to classify the inbound packets and assign them appropriate TP levels.

14 Claims, 10 Drawing Sheets



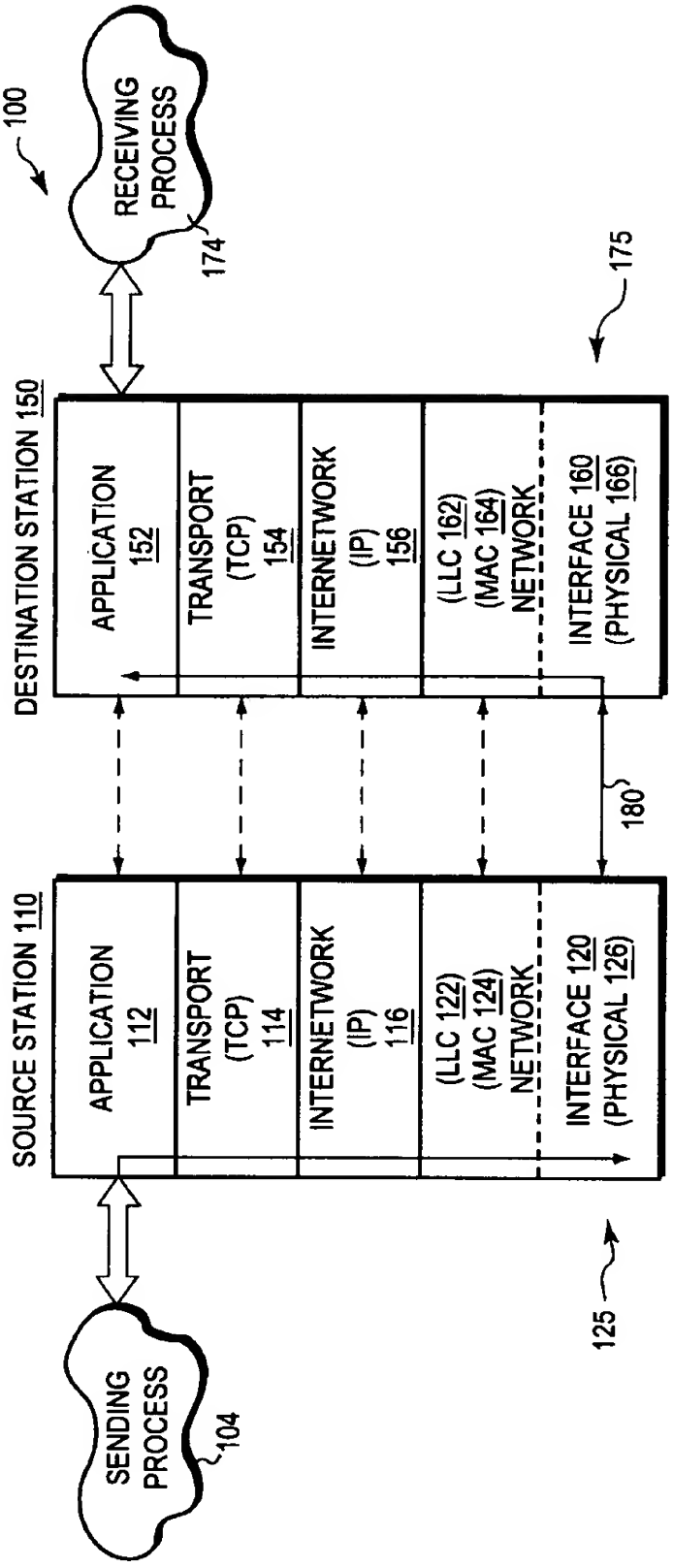


FIG. 1
(PRIOR ART)

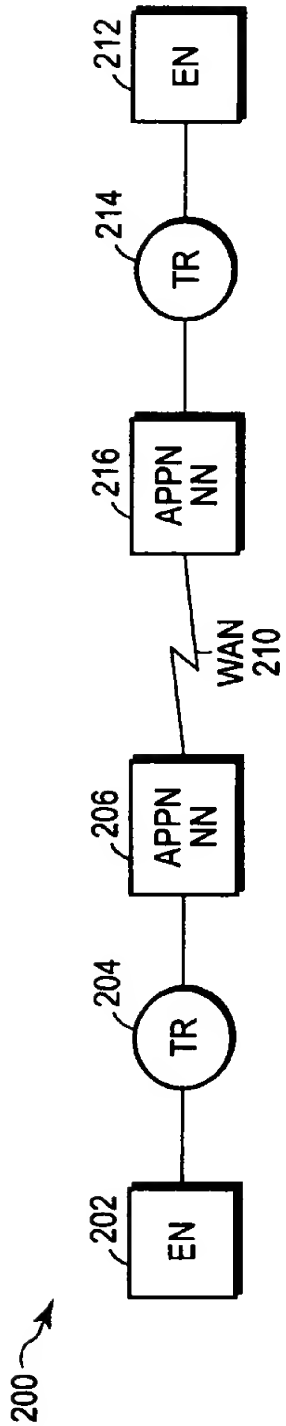


FIG. 2
(PRIOR ART)

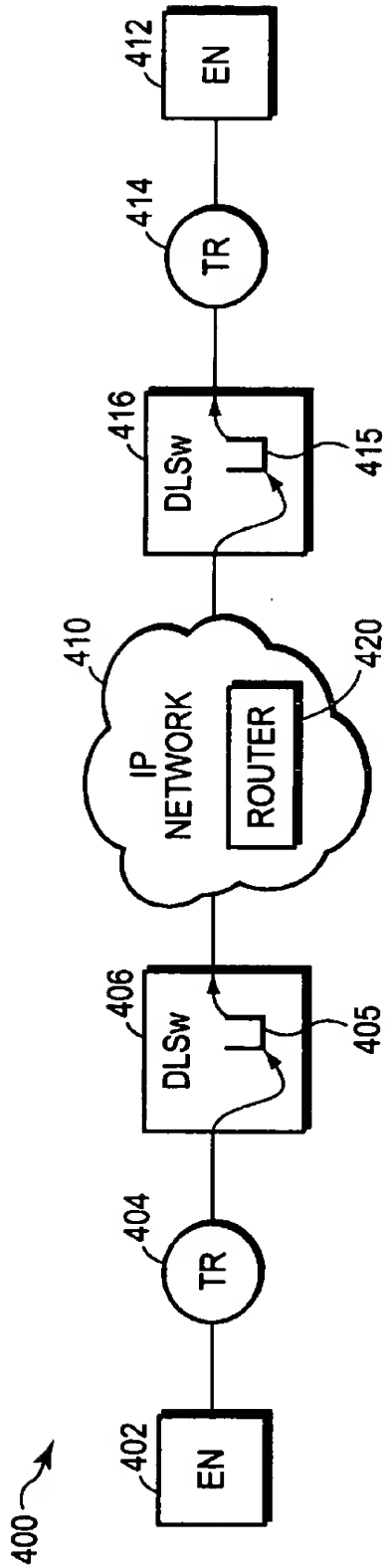


FIG. 4
(PRIOR ART)

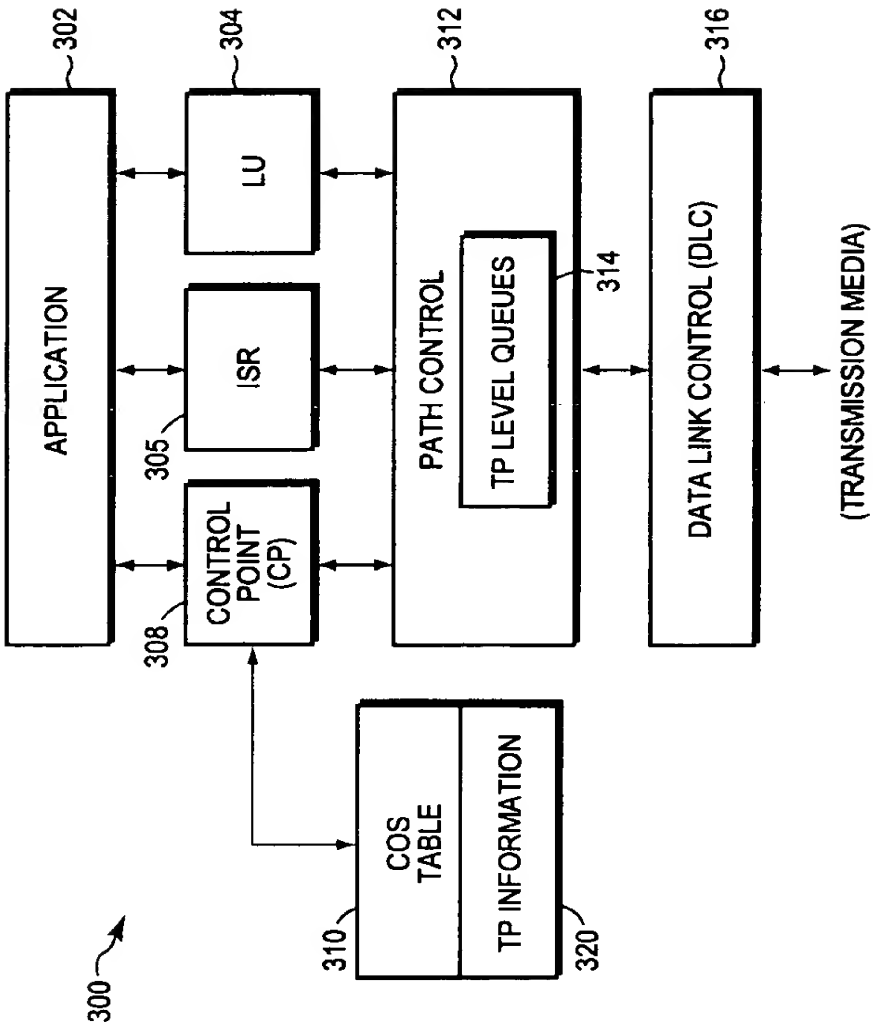


FIG. 3
(PRIOR ART)

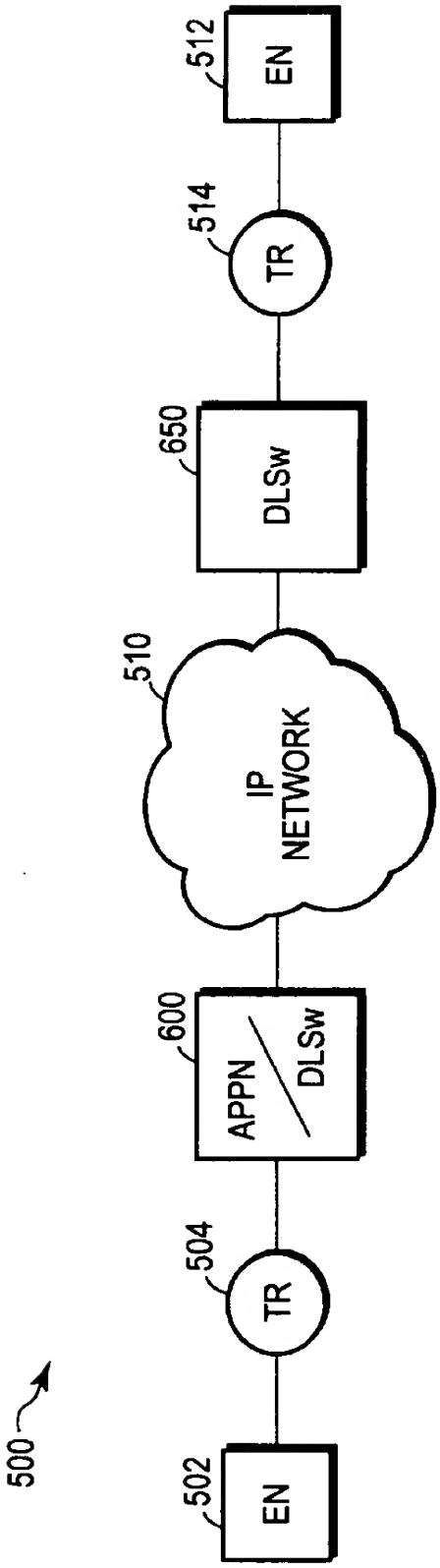

500
502
EN
504
TR
600
APPN / DLSw
510
IP NETWORK
650
DLSw
514
TR
512
EN

FIG. 5

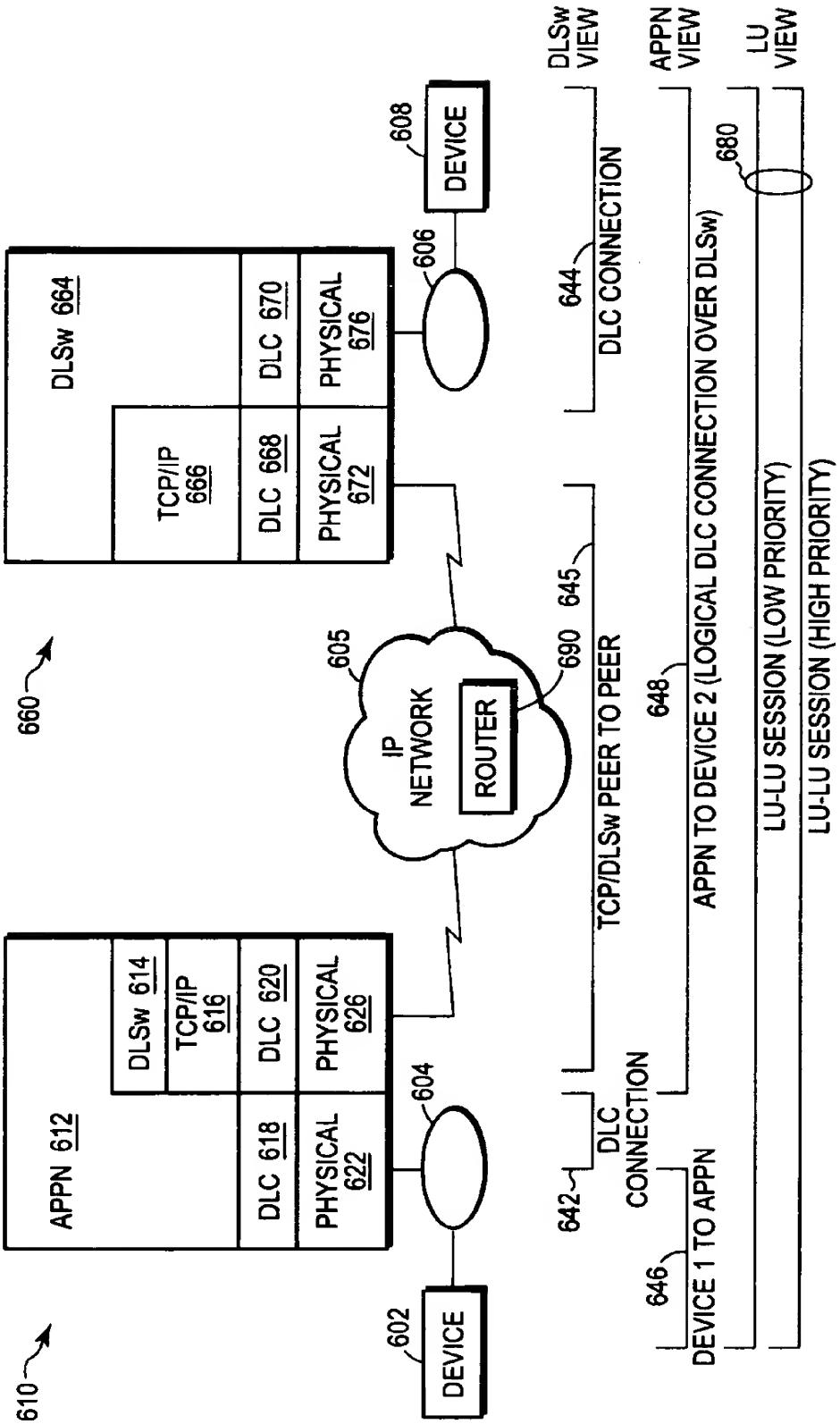

APPN/DLSw 600
APPN 612
DLSw 614
TCP/IP 616
DLC 618
DLC 620
PHYSICAL 622
PHYSICAL 626
604
602
DEVICE
610
660
DLSw NODE 650
DLSw 664
TCP/IP 666
DLC 668
DLC 670
PHYSICAL 672
PHYSICAL 676
606
608
DEVICE
605
IP NETWORK
ROUTER
690
645
TCP/DLSw PEER TO PEER
642
DLC CONNECTION
644
DLC CONNECTION
DLSw VIEW
APPN VIEW
LU VIEW
648
APPN TO DEVICE 2 (LOGICAL DLC CONNECTION OVER DLSw)
680
LU-LU SESSION (LOW PRIORITY)
LU-LU SESSION (HIGH PRIORITY)
646
DEVICE 1 TO APPN

FIG. 6

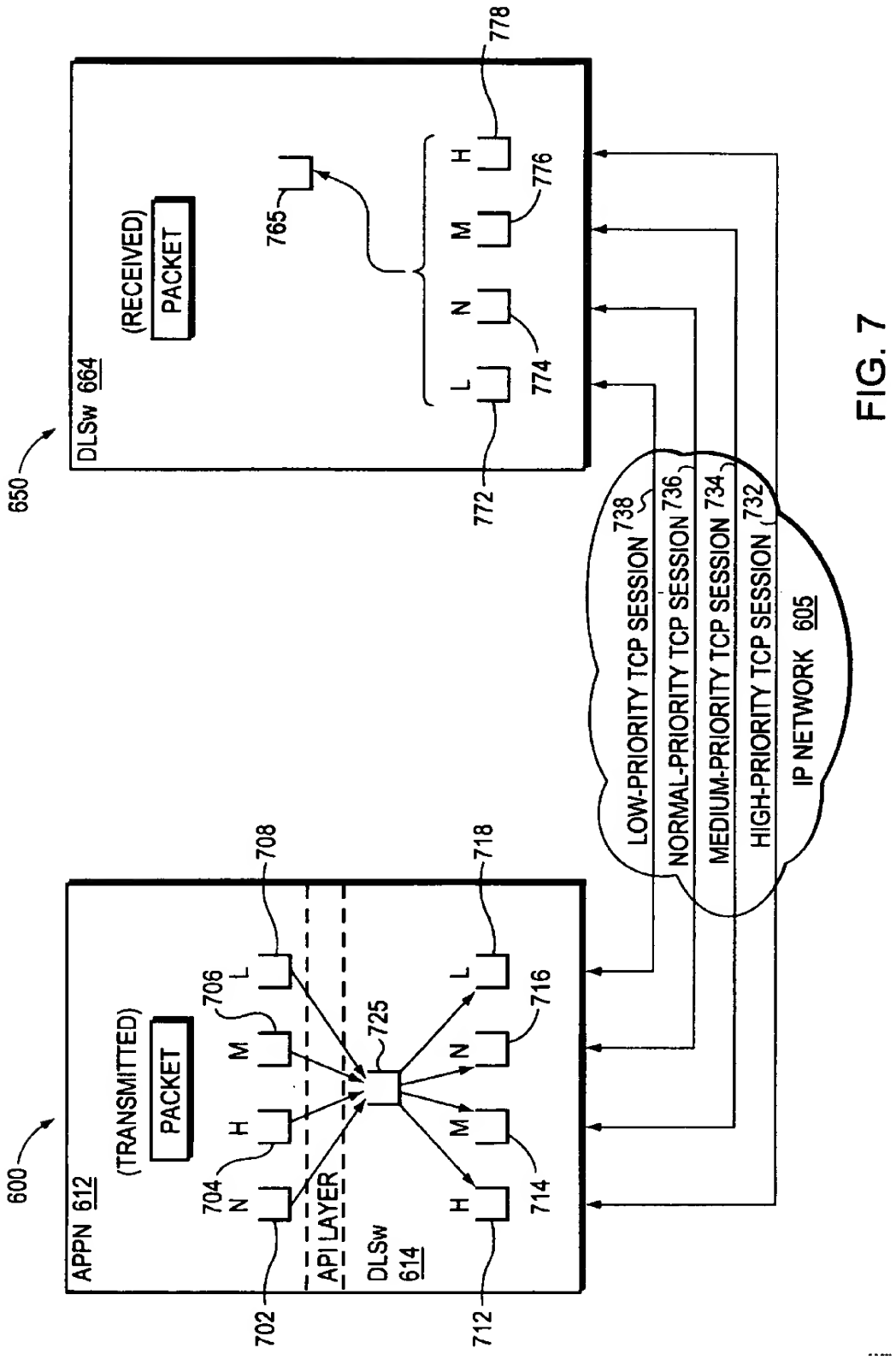

600
APPN 612
(TRANSMITTED)
PACKET
704
N
H
M
706
L
API LAYER
DLSw 614
725
708
718
L
M
N
H
716
714
702
712
650
DLSw 664
(RECEIVED)
PACKET
765
L
N
M
H
774
776
778
772
LOW-PRIORITY TCP SESSION 738
NORMAL-PRIORITY TCP SESSION 736
MEDIUM-PRIORITY TCP SESSION 734
HIGH-PRIORITY TCP SESSION 732
IP NETWORK 605

FIG. 7

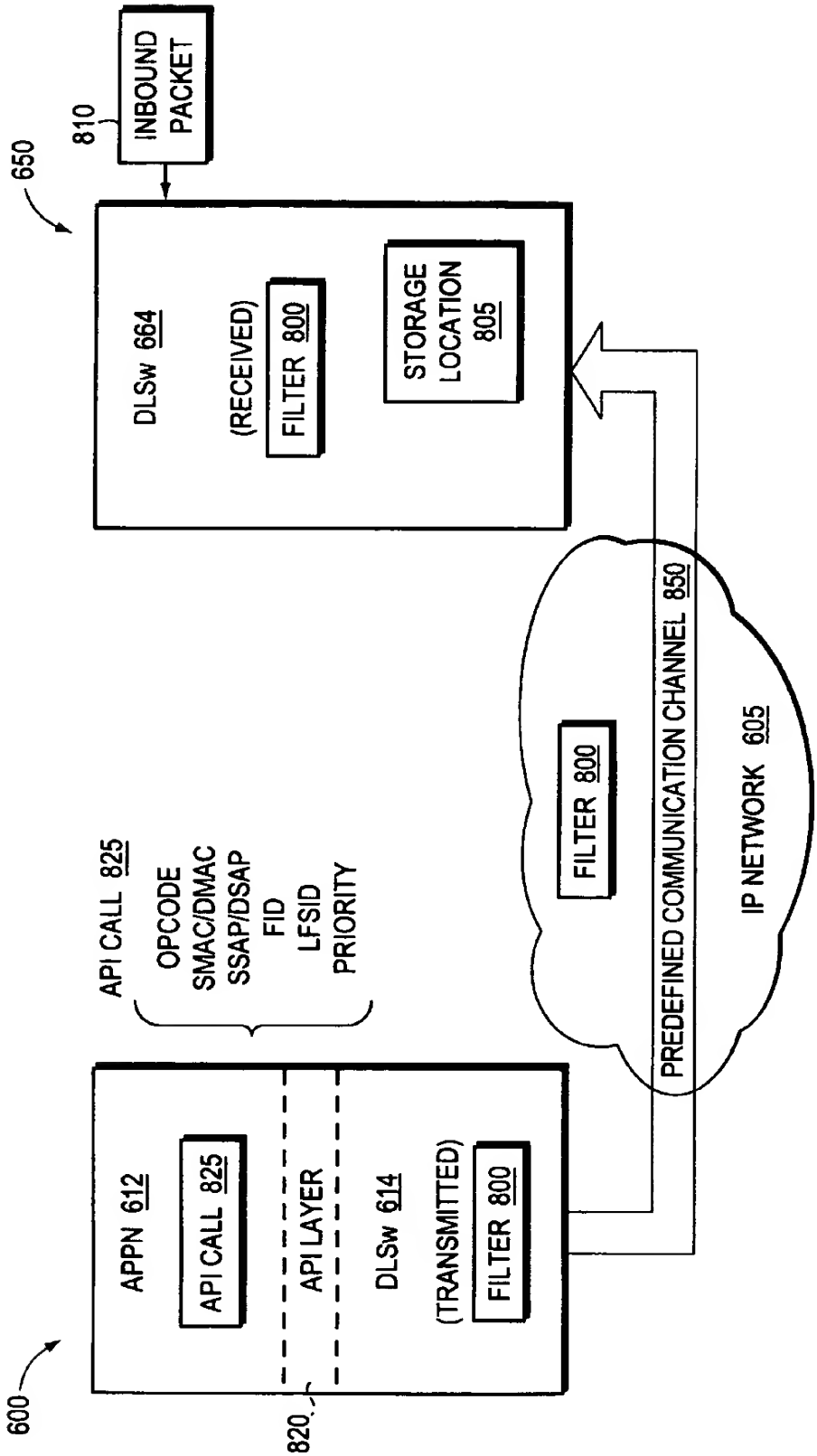

600
650
810
INBOUND PACKET
DLSw 664
(RECEIVED)
FILTER 800
STORAGE LOCATION 805
PREDEFINED COMMUNICATION CHANNEL 850
IP NETWORK 605
FILTER 800
API CALL 825
OPCODE
SMAC/DMAC
SSAP/DSAP
FID
LFSID
PRIORITY
APPN 612
API CALL 825
API LAYER
820
DLSw 614
(TRANSMITTED)
FILTER 800

FIG. 8

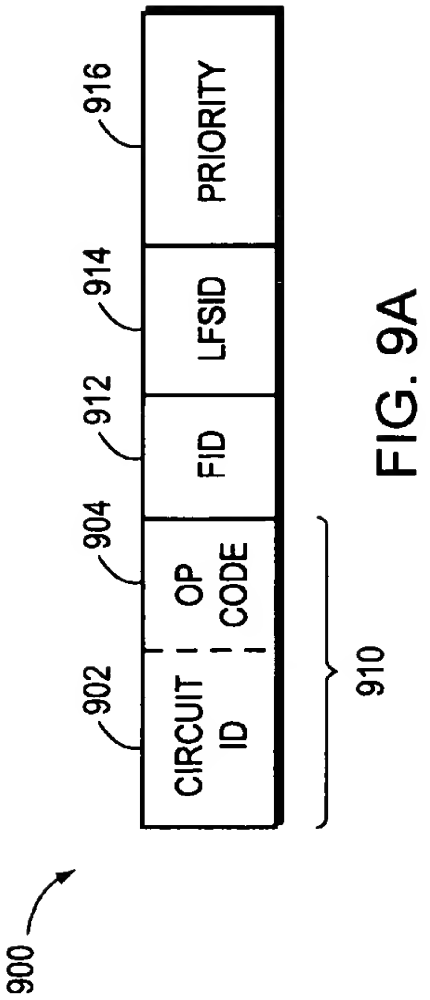

900
902
CIRCUIT ID
904
OP CODE
912
FID
914
LFSID
916
PRIORITY
910

FIG. 9A

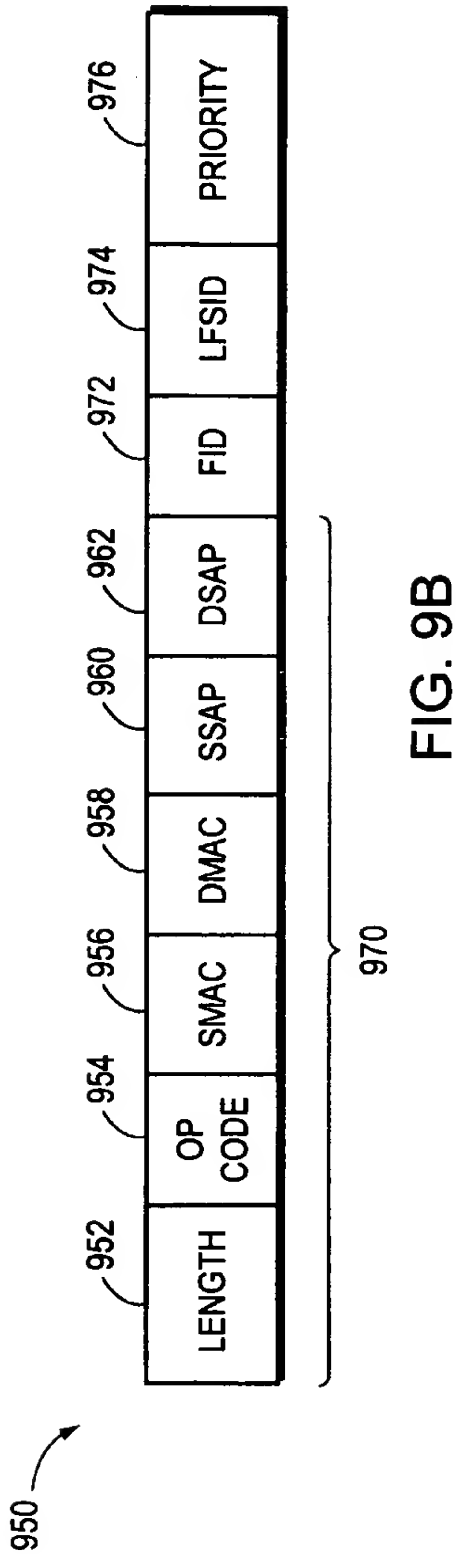

950
952
LENGTH
954
OP CODE
956
SMAC
958
DMAC
960
SSAP
962
DSAP
972
FID
974
LFSID
976
PRIORITY
970

FIG. 9B

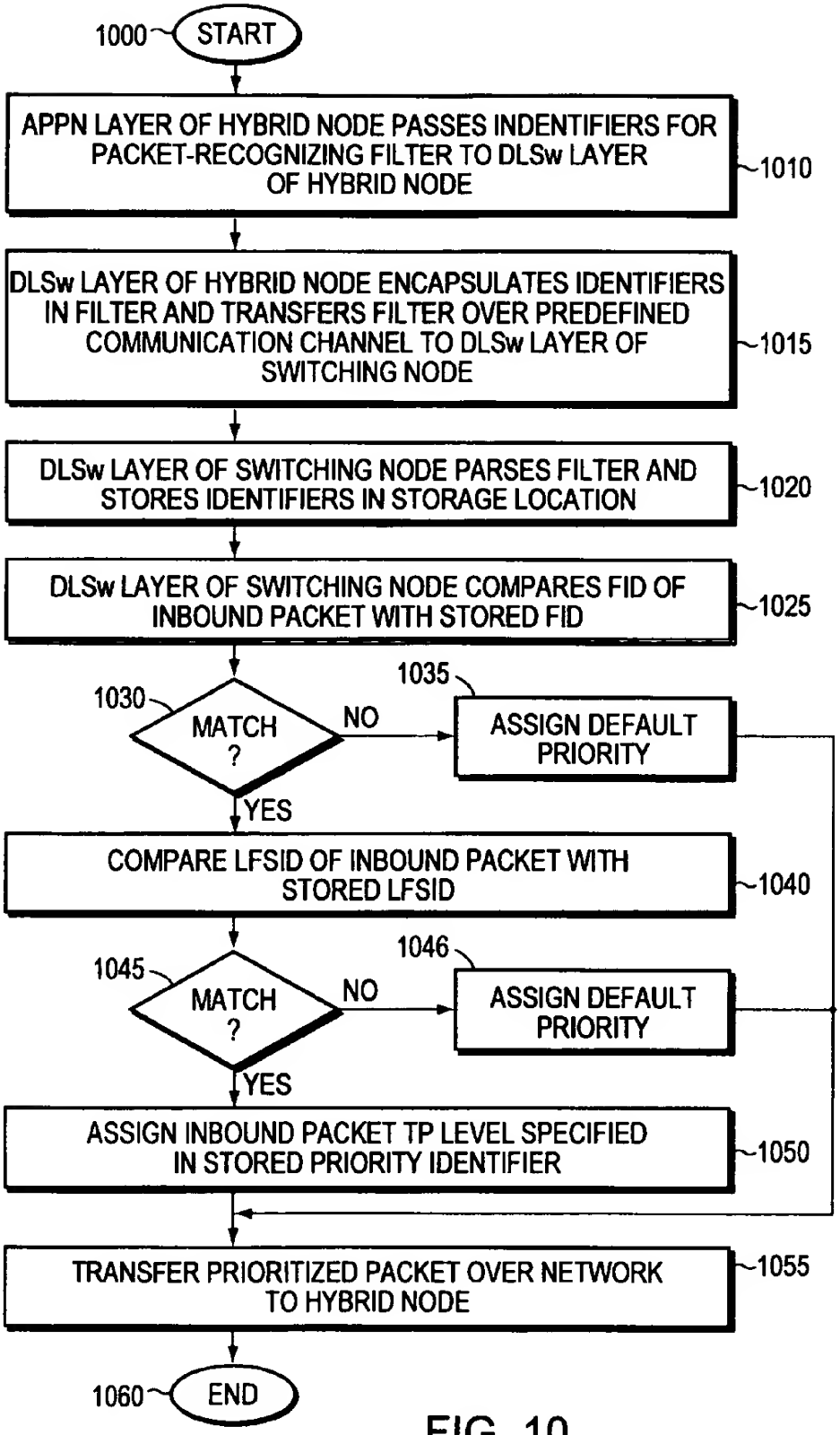

1000
START
APPN LAYER OF HYBRID NODE PASSES INDENTIFIERS FOR PACKET-RECOGNIZING FILTER TO DLSw LAYER OF HYBRID NODE
1010
DLSw LAYER OF HYBRID NODE ENCAPSULATES IDENTIFIERS IN FILTER AND TRANSFERS FILTER OVER PREDEFINED COMMUNICATION CHANNEL TO DLSw LAYER OF SWITCHING NODE
1015
DLSw LAYER OF SWITCHING NODE PARSES FILTER AND STORES IDENTIFIERS IN STORAGE LOCATION
1020
DLSw LAYER OF SWITCHING NODE COMPARES FID OF INBOUND PACKET WITH STORED FID
1025
1030
MATCH ?
NO
1035
ASSIGN DEFAULT PRIORITY
YES
COMPARE LFSID OF INBOUND PACKET WITH STORED LFSID
1040
1045
MATCH ?
NO
1046
ASSIGN DEFAULT PRIORITY
YES
ASSIGN INBOUND PACKET TP LEVEL SPECIFIED IN STORED PRIORITY IDENTIFIER
1050
TRANSFER PRIORITIZED PACKET OVER NETWORK TO HYBRID NODE
1055
1060
END

FIG. 10

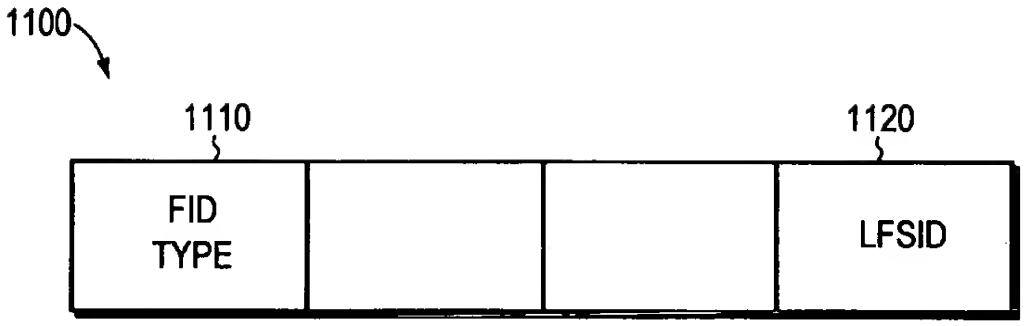


FIG. 11
(PRIOR ART)

TECHNIQUE FOR CAPTURING INFORMATION NEEDED TO IMPLEMENT TRANSMISSION PRIORITY ROUTING AMONG HETEROGENEOUS NODES OF A COMPUTER NETWORK

CROSS-REFERENCE TO RELATED APPLICATIONS

This application is a continuation of U.S. patent application Ser. No. 08/833,834 filed Apr. 10, 1997, now U.S. Pat. No. 5,940,390 titled "MECHANISM FOR CONVEYING DATA PRIORITIZATION INFORMATION AMONG HETEROGENEOUS NODES OF A COMPUTER NETWORK." The entirety of the disclosure of said copending application is incorporated herein by reference.

This invention is related to the following copending U.S. Patent Applications:

U.S. patent application Ser. No. 08/839,435, titled TECHNIQUE FOR MAINTAINING PRIORITIZATION OF DATA TRANSFERRED AMONG HETEROGENEOUS NODES OF A COMPUTER NETWORK; patent application Ser. No. 08/833,837, titled "TECHNIQUE FOR CAPTURING INFORMATION NEEDED TO IMPLEMENT TRANSMISSION PRIORITY ROUTING AMONG HETEROGENEOUS NODES OF A COMPUTER NETWORK," which applications were filed on Apr. 10, 1997 and assigned to the assignee of the present invention. U.S. patent application Ser. No. 08/926,539, titled "TECHNIQUE FOR REDUCING THE FLOW OF TOPOLOGY INFORMATION AMONG NODES OF A COMPUTER NETWORK," which application was filed on Sep. 10, 1997 and assigned to the assignee of the present invention.

FIELD OF THE INVENTION

The invention relates to computer networks and, more particularly, to the distribution of packet prioritization information among stations of a computer network.

BACKGROUND OF THE INVENTION

Data communication in a computer network involves the exchange of data between two or more entities interconnected by communication links and subnetworks. These entities are typically software programs executing on hardware computer platforms, such as end stations and intermediate stations. Examples of an intermediate station may be a router or switch which interconnects the communication links and subnetworks to enable transmission of data between the end stations. A local area network (LAN) is an example of a subnetwork that provides relatively short distance communication among the interconnected stations; in contrast, a wide area network (WAN) enables long distance communication over links provided by public or private telecommunications facilities.

Communication software executing on the end stations correlate and manage data communication with other end stations. The stations typically communicate by exchanging discrete packets or frames of data according to predefined protocols. In this context, a protocol consists of a set of rules defining how the stations interact with each other. In addition, network routing software executing on the routers allow expansion of communication to other end stations. Collectively, these hardware and software components comprise a communications network and their interconnections are defined by an underlying architecture.

Modern communications network architectures are typically organized as a series of hardware and software levels

or "layers" within each station. These layers interact to format data for transfer between, e.g., a source station and a destination station communicating over the network. Specifically, predetermined services are performed on the data as it passes through each layer and the layers communicate with each other by means of the predefined protocols. The lower layers of these architectures are generally standardized and are typically implemented in hardware and firmware, whereas the higher layers are generally implemented in the form of software running on the stations attached to the network. Examples of such communications architectures include the Systems Network Architecture (SNA) developed by International Business Machines Corporation and the Internet communications architecture.

The Internet architecture is represented by four layers which are termed, in ascending interfacing order, the network interface, internetwork, transport and application layers. These layers are arranged to form a protocol stack in each communicating station of the network. FIG. **1** illustrates a schematic block diagram of prior art Internet protocol stacks **125** and **175** used to transmit data between a source station **110** and a destination station **150**, respectively, of a network **100**. As can be seen, the stacks **125** and **175** are physically connected through a communications channel **180** at the network interface layers **120** and **160**. For ease of description, the protocol stack **125** will be described.

In general, the lower layers of the communications stack provide internetworking services and the upper layers, which are the users of these services, collectively provide common network application services. The application layer **112** provides services suitable for the different types of applications using the network, while the lower network interface layer **120** of the Internet architecture accepts industry standards defining a flexible network architecture oriented to the implementation of LANs.

Specifically, the network interface layer **120** comprises physical and data link sublayers. The physical layer **126** is concerned with the actual transmission of signals across the communication channel and defines the types of cabling, plugs and connectors used in connection with the channel. The data link layer, on the other hand, is responsible for transmission of data from one station to another and may be further divided into two sublayers: Logical Link Control (LLC **122**) and Media Access Control (MAC **124**).

The MAC sublayer **124** is primarily concerned with controlling access to the transmission medium in an orderly manner and, to that end, defines procedures by which the stations must abide in order to share the medium. In order for multiple stations to share the same medium and still uniquely identify each other, the MAC sublayer defines a hardware or data link address called a MAC address. This MAC address is unique for each station interfacing to a LAN. The LLC sublayer **122** manages communications between devices over a single link of the network and provides for environments that need connectionless or connection-oriented services at the data link layer.

Connection-oriented services at the data link layer generally involve three distinct phases: connection establishment, data transfer and connection termination. During connection establishment, a single path is established between the source and destination stations. This connection, e.g., an IEEE 802.2 LLC Type 2 or "Data Link Control" (DLC) connection as referred hereinafter, is based on the use of service access points (SAPs); a SAP is generally the address of a port or access point to a higher-

level layer of a station. Once the connection has been established, data is transferred sequentially over the path and, when the DLC connection is no longer needed, the path is terminated. The details of such connection establishment and termination are well-known and, thus, will not be described herein.

The transport layer **114** and the internetwork layer **116** are substantially involved in providing predefined sets of services to aid in connecting the source station to the destination station when establishing application-to-application communication sessions. The primary network layer protocol of the Internet architecture is the Internet protocol (IP) contained within the internetwork layer **116**. IP is primarily a connectionless network protocol that provides internetwork routing, fragmentation and reassembly of datagrams and that relies on transport protocols for end-to-end reliability. An example of such a transport protocol is the Transmission Control Protocol (TCP) contained within the transport layer **114**. Notably, TCP provides connection-oriented services to the upper layer protocols of the Internet architecture. The term TCP/IP is commonly used to refer to the Internet architecture.

Data transmission over the network **100** therefore consists of generating data in, e.g., sending process **104** executing on the source station **110**, passing that data to the application layer **112** and down through the layers of the protocol stack **125**, where the data are sequentially formatted as a frame for delivery onto the channel **180** as bits. Those frame bits are then transmitted over an established connection of channel **180** to the protocol stack **175** of the destination station **150** where they are passed up that stack to a receiving process **174**. Data flow is schematically illustrated by solid arrows.

Although actual data transmission occurs vertically through the stacks, each layer is programmed as though such transmission were horizontal. That is, each layer in the source station **110** is programmed to transmit data to its corresponding layer in the destination station **150**, as schematically shown by dotted arrows. To achieve this effect, each layer of the protocol stack **125** in the source station **110** typically adds information (in the form of a header field) to the data frame generated by the sending process as the frame descends the stack. At the destination station **150**, the various encapsulated headers are stripped off one-by-one as the frame propagates up the layers of the stack **175** until it arrives at the receiving process.

SNA is a mainframe-oriented network architecture that also uses a layered approach. The services included within this architecture are generally similar to those defined in the Internet communications architecture. In a SNA network, though, applications executing on end stations typically access the network through logical units (LU) of the stations; accordingly, in a typical SNA network, a communication session connects two LUs in a LU—LU session. Activation and deactivation of such a session is addressed by Advanced Peer to Peer Networking (APPN) functions.

The APPN functions generally include session establishment and session routing within an APPN network. FIG. **2** is a schematic block diagram of a prior art APPN network **200** comprising two end stations **202, 212**, which are typically configured as end nodes (EN), coupled to token ring (TR) subnetworks **204, 214**, respectively. During session establishment, an EN (such as EN **202**) requests an optimum route for a session between two LUs; this route is calculated and conveyed to EN **202** by an intermediate station functioning as a network node server (e.g., station **206**) via a LOCATE message exchange through the network **200**. Thereafter, a "set-up" or BIND message is forwarded over the route to initiate the session. The BIND includes information pertaining to the partner LU requested for the session.

Intermediate session routing occurs when the intermediate stations **206, 216**, configured as APPN network nodes (NN), are present in a session between the two end nodes. As can be seen, the APPN network nodes are further interconnected by a WAN **210** that extends the APPN architecture throughout the network. The APPN network nodes forward packets of an LU—LU session over the calculated route between the two APPN end nodes. An APPN network node is a full-functioning APPN router node having all APPN base service capabilities, including session services functions. An APPN end node, on the other hand, is capable of performing only a subset of the functions provided by an APPN network node. APPN network and end nodes are well-known and are, for example, described in detail in *Systems Network Architecture Advanced Peer to Peer Networking Architecture Reference* IBM Doc SC30-3422 and *APPN Networks* by Jesper Nilausen, printed by John Wiley and Sons, 1994, at pgs 11–83.

FIG. **3** is a schematic block diagram of the software architecture of a prior art APPN node **300**. As noted, application **302** executing on an APPN end node, such as EN **202** of network **200**, communicates with another end node, such as EN **212**, through a LU—LU session; LU **304** within each end node functions as both a logical port for the application to the network and as an end point of the communication session. The session generally passes through a path control module **312** and a data link control (DLC) module **316** of the node, the latter of which connects to various network transmission media.

When functioning as an APPN router node, such as NN **206**, an intermediate session routing (ISR) module **305** maintains a portion of the session in each "direction" with respect to an adjacent network node, such as NN **216** of network **200**. In response to receiving the BIND message during session establishment, path control **312** and ISR **305** are invoked to allocate resources for the session. In particular, each NN **206, 216** allocates a local form session identifier (LFSID) for each direction of the session; the LFSID is thereafter appended to the packets in a SNA transmission header (TH) to identify the session context. Collectively, each of these individually-established "local" sessions form the logical communication session between the LUs **304** of the end nodes **202, 212**.

When initiating a session, the application **302** specifies a mode name that is carried within the BIND message and distributed to all APPN network nodes; the LU **304** in each node uses the mode name to indicate the set of required characteristics for the session being established. Specifically, the mode name is used by control point (CP) module **308** of each APPN node **300** to find a corresponding class of service (COS) as defined in a COS table **310**. The CP coordinates performance of all APPN functions within the node, including management of the COS table **310**. The COS definition in table **310** includes a priority level specified by transmission priority (TP) information **320** for the packets transferred over the session; as a result, each APPN network node is apprised of the priority associated with the packets of a LU—LU session. The SNA architecture specifies four (4) TP levels: network priority, high priority, medium priority and low priority. Path control **312** maintains a plurality of queues **314**, one for each TP level, for transmitting packets onto the transmission media via DLC **316**.

Data link switching (DLSw) is a forwarding mechanism for the SNA architecture over an IP backbone network, such as the Internet. A heterogeneous DLSw network is formed when two DLSw switches interconnect the end nodes of the APPN network by way of the IP network; the DLSw switches preferably communicate using a switch-to-switch protocol (SSP) that provides packet "bridging" operations at the LLC (i.e., DLC) protocol layer. FIG. 4 is a schematic block diagram of a prior art DLSw network **400** comprising DLSw switches **406, 416** interconnecting ENs **402, 412** via IP network **410**. The DLSw forwarding mechanism is also well-known and described in detail in *Request for Comment* (*RFC*) 1795 by Wells & Bartky, 1995 at pgs 1–91.

According to the DLSw technique, a lower-layer DLC connection is established between each EN and DLSw switch; however, these connections terminate at the switches **406, 416**. In order to provide a complete end-to-end connection between the end nodes, the DLC connections are "disposed" over a reliable, higher-layer transport mechanism, such as TCP sessions. DLSw switches can establish multiple, parallel TCP sessions using well-known port numbers. All packets associated with a particular DLC connection typically follow a single, designated TCP session. Accordingly, SNA data frames originating at a sending EN **402** are transmitted over a particular DLC connection along TR **404** to DLSw switch **406**, where they are encapsulated within a designated TCP session as packets and transported over IP network **410**. The packets are received by DLSw switch **416**, decapsulated to their original frames and transmitted over a corresponding DLC connection of TR **414** to EN **412** in the order received by switch **406** from EN **402**.

Typically, all packets transmitted by DLSw switch **406** over a DLC connection/TCP session flow at the same priority level from a single output queue **405** of the switch and arrive at an output queue **415** of DLSw switch **416** in the same order in which they are transmitted. When the switches are configured as bridges to forward packets over a TCP session through the IP network, prioritization is straightforward. However, it may be desired to integrate the functions of an APPN network node within switch **406** by overlaying an APPN layer onto a DLSw layer of the switch; the resulting hybrid node may prioritize the packets at the APPN layer in an order governed by the TP information levels.

A problem that arises when deploying a hybrid node in such a heterogeneous network is that the TP priority information is lost when passing the packets between the APPN and DLSw layers, primarily because the TP information is not encapsulated within the packets. That is, the APPN layer has knowledge of the TP levels associated with the packets of a LU—LU session as a result of the BIND message exchange during session establishment; yet that information is not encapsulated within the associated packets and, thus, is not conveyed beyond the APPN layer. An example of a tagging mechanism suitable for use with the present invention that conveys TP levels from the APPN layer to the DLSw layer is disclosed in copending and commonly-assigned U.S. patent application, titled *Technique for Maintaining Prioritization of Data Transferred Among Heterogeneous Nodes of a Computer Network*, filed herewith and incorporated by reference as though fully set forth herein.

As described in the commonly-assigned application, the APPN protocol layer of the hybrid node assigns a TP level to each packet and passes that priority information to the DLSw layer of the node via an application programming interface extension. The TP level is converted to information that is "tagged" to each packet and the DLSw layer allocates each tagged packet to a TCP session based on the assigned TP level. The tagged information is then encapsulated within an IP header to enable intermediate routers to maintain the order and priority of the packet as it is transmitted outbound over the IP network to a receiving DLSw switch.

However, the tagged information within the IP header is not discernible to the receiving DLSw switch and, thus, the switch has no knowledge of the TP level associated with the outbound packet. If that packet requests a response, the DLSw switch cannot select, on the basis of priority, the proper TCP session over which to transmit a corresponding inbound packet; accordingly, the switch arbitrarily chooses a session. If the chosen TCP session has a lower designated priority than the session carrying the outbound packet, network throughput may be negatively impacted.

One solution to this problem is to deploy another hybrid node in place of the receiving DLSw switch. This approach is undesirable primarily because a goal of heterogeneous network design is to minimize the number of hybrid nodes in the network. A reason for minimizing the number of hybrid nodes is that such nodes require additional processing and memory resources, thereby resulting in expensive deployments. The present invention is directed to solving the problem of distributing packet prioritization information, assigned by a hybrid node of a heterogeneous network, to switching nodes of the network.

SUMMARY OF THE INVENTION

The invention comprises a mechanism for conveying information pertaining to transmission priority (TP) levels of inbound packets transmitted over a heterogeneous network from a switching node to a hybrid node of the network. The mechanism comprises a packet-recognizing filter having a novel format that is generated by the hybrid node and dynamically transmitted to the switching node over a predefined communication channel of the network. As described further herein, the filter enables the switching node to classify the inbound packets and assign them appropriate TP levels.

In the illustrative embodiment, the heterogeneous network is preferably a data link switching (DLSw) network with end nodes interconnected by way of an Internet protocol (IP) backbone network and the hybrid node is an advanced peer-to-peer networking (APPN) node with DLSw capabilities. Applications executing on the end nodes communicate via logical unit to logical unit (LU—LU) sessions, whereas the switching node communicates with the APPN node using a switch-to-switch protocol (SSP) over data link control (DLC) connections associated with the LU—LU sessions of the DLSw network; these DLC connections are further overlayed onto existing transmission control protocol (TCP) sessions of the IP network. Preferably, each TCP session is further associated with a TP level.

According to aspects of the invention, the predefined communication channel may be implemented as either an in-band channel over one of the existing TCP sessions using novel extensions to SSP, or an out-band channel over a newly-created TCP session. The format of the filter is preferably customized for each channel implementation; nevertheless, each filter includes a unique opcode identifying the filter, a format identifier (FID) denoting the format of a specific inbound packet, a local form session identifier (LFSID) that classifies the LU—LU session context of the specific packet and a priority identifier specifying the TP level of the packet.

Operationally, an APPN protocol layer of the APPN node passes the opcode, LFSID, FID and priority identifier to a

DLSw protocol layer of the node, through an application programming interface (API), during establishment of the LU—LU session. In response to the API, the DLSw layer encapsulates these identifiers within fields of the filter and transfers the filter over the communication channel to the switching node. When transferring the filter over the in-band communication channel, the opcode is encapsulated within a SSP header, whereas for the out-band channel embodiment, additional addressing information is encapsulated with the opcode in fields of a defined header.

Upon receiving the filter, a DLSw layer of the switching node stores the LFSID, FID and priority identifier and proceeds to examine each inbound packet prior to forwarding it to the APPN node. Specifically, the switching node initially determines the format of each packet and if it matches the stored FID, the node compares the LFSID of the inbound packet with the stored LFSID to identify the LU—LU session context of the packet. If the values of these latter identifiers match, the switching node assigns to the inbound packet the TP level specified by the stored priority identifier and forwards the packet to the APPN node over an appropriate one of the existing TCP sessions.

BRIEF DESCRIPTION OF THE DRAWINGS

The above and further advantages of the invention may be better understood by referring to the following description in conjunction with the accompanying drawings in which like reference numbers indicate identical or functionally similar elements:

FIG. **1** is a schematic block diagram of prior art communications architecture protocol stacks, such as the Internet protocol stack, used to transmit data between stations of a computer network;

FIG. **2** is a schematic block diagram of a prior art Advanced Peer to Peer Networking (APPN) network including APPN nodes;

FIG. **3** is a schematic block diagram of the software architecture a prior art APPN node;

FIG. **4** is a schematic block diagram of a prior art data link switching (DLSw) network;

FIG. **5** is a block diagram of a heterogeneous computer network, including a DLSw node and an APPN/DLSw hybrid node for interconnecting various subnetworks and communication links on which the present invention may advantageously operate;

FIG. **6** is a schematic block diagram of protocol stacks contained within the DLSw and APPN/DLSw nodes of FIG. **5**;

FIG. **7** is a schematic block diagram illustrating the assignment of priority levels among established communication sessions and the distribution of packets among the sessions;

FIG. **8** is a schematic block diagram of a novel packet-recognizing filter generated by the hybrid node of FIG. **5** and dynamically transmitted to the DLSw node over a predefined communication channel in accordance with the invention;

FIGS. **9**A and **9**B are schematic block diagrams depicting formats of the novel packet-recognizing filter of FIG. **8**;

FIG. **10** is a flowchart illustrating use of the novel filter in accordance with the present invention; and

FIG. **11** is a schematic block diagram depicting the format of a conventional transmission header upon which the inventive packet-recognizing filter may advantageously operate.

DETAILED DESCRIPTION OF ILLUSTRATIVE EMBODIMENT

FIG. **5** is a block diagram of a computer network **500** comprising a collection of interconnected communication links and subnetworks attached to a plurality of stations. The stations are typically computers comprising end stations **502, 512** and intermediate stations **600, 650**. Preferably, the end stations are Advanced Peer to Peer Networking (APPN) end nodes, although the stations may comprise other types of nodes such as Low Entry Networking nodes or Physical Units 2.0 via Dependent Logical Unit Requestor functions. In addition, the intermediate station **650** is a data link switching (DLSw) node and intermediate station **600** is an APPN/DLSw hybrid node.

Each node typically comprises a plurality of interconnected elements, such as a processor, a memory and a network adapter. The memory may comprise storage locations addressable by the processor and adapter for storing software programs and data structures associated with the inventive filtering mechanism and techniques. The processor may comprise processing elements or logic for executing the software programs and manipulating the data structures. An operating system, portions of which are typically resident in memory and executed by the processor, functionally organizes the node by, inter alia, invoking network operations in support of software processes executing on the node. It will be apparent to those skilled in the art that other processor and memory means, including various computer readable media, may be used for storing and executing program instructions pertaining to the techniques described herein.

The subnetworks included within network **500** are preferably local area networks (LANs) and the communication links may include wide area network (WAN) links; in the illustrative embodiment of the invention, the LANs are preferably token rings (TR) **504, 514** and an IP network **510**, which may comprise either a LAN and/or a WAN configuration such as X.25, interconnects the nodes **600, 650**. Communication among the nodes coupled to the network **500** is typically effected by exchanging discrete data packets or frames via connection-oriented service sessions between the communicating nodes.

Heterogeneous (DLSw) network **500** is formed when APPN/DLSw hybrid node **600** is connected to DLSw node **650** via IP network **510**. FIG. **6** is a schematic block diagram of protocol stacks **610, 660** within the nodes **600** and **650**, respectively. Applications executing on SNA devices (end stations) **602, 608** typically access the network through logical units (LUs) of the stations and communicate via LU—LU sessions. Hybrid node **600** functions to facilitate establishment and routing of these connection-oriented communication sessions within the network. To this end, protocol stack **610** preferably comprises an APPN layer **612** that contains the software modules described in FIG. **3**.

The stack **610** also includes a Transmission Control Protocol/Internet protocol (TCP/IP) layer **616** containing those layers of the Internet communications architecture protocol stack (FIG. **1**) needed to establish, e.g., conventional connection-oriented, TCP communication sessions. Physical sublayers **622** and **626** specify the electrical, mechanical, procedural and functional specifications for activating, maintaining and de-activating the physical links **604** and **605** of the network. Protocol stack **660** of DLSw node **650** likewise includes a TCP/IP layer **666** and physical sublayers **672** and **676**, which are functionally equivalent to those layers of protocol stack **610**.

Each node **600, 650** further contains a DLSw layer **614, 664** and data link control (DLC) layers **618, 620** and **668,**

670, respectively, the latter layers providing a connection-oriented service via conventional DLC connections. The DLSw layers provide a mechanism for forwarding data frame traffic between devices 602, 608 over IP network 605. Preferably, the DLSw layers 614, 664 cooperate in a peer-relationship and communicate via a switch-to-switch protocol (SSP) to, inter alia, define TCP sessions over the IP network.

In the illustrative embodiment, there are a plurality of connection/session "views" established within the network. For example, from an APPN view, there is a DLC connection 646 between device 602 and APPN layer 612 of node 600, and a DLC connection 648 between APPN layer 612 and device 608. From a DLSw view, there is a DLC connection 642 between APPN layer 612 and DLSw layer 614 of node 600, and a DLC connection 644 between DLSw layer 664 and device 608; in order to provide reliable, end-to-end connections between the devices, these DLC connections are "overlayed" onto TCP sessions (denoted 645) between the two DLSw layers 614, 664. Lastly, from a LU view, there are multiple LU—LU sessions 680 (at various priority levels) between the LUs of devices 602 and 608.

It should be noted that the TCP sessions are initiated between DLSw peers 614, 664 in accordance with a conventional TCP transport protocol. Thereafter, SSP control messages are exchanged between the DLSw layers 614, 664 of the nodes to establish an end-to-end DLSw circuit over the session. Information contained within these control messages are used to generate a DLSw circuit identifier (ID) that associates the DLSw circuit with the session. Preferably, the DLC connections 642, 644 overlayed on the TCP session 645 "map" to the DLSw circuit. The generation of DLSw circuits and identifiers is described in *Request for Comment* (*RFC*) 1795 by Wells & Bartky, 1995, while the establishment of multiple TCP sessions between DLSw peer layers is described in both *RFC* 1795 and *Internetworking with TCP/IP* by Comet and Stevens, printed by Prentice Hall, 1991; all of these publications are hereby incorporated by reference as though fully set forth herein.

Typically, packets transmitted by a DLSw switch over a TCP session flow at the same priority level from a single output queue of the switch and arrive at a peer DLSw switch in the same order in which they are transmitted. Hybrid node 600 may, however, prioritize the packets of a LU—LU session at the APPN layer 612 in an order specified by transmission priority (TP) information contained within the node 600. FIG. 7 is a schematic block diagram illustrating the assignment of TP levels among established communication sessions and the distribution of packets among those sessions.

A path control module 312 (FIG. 3) of the APPN layer 612 within node 600 maintains four queues 702–708, one for each TP level, for transmitting data packets (received from DLC connection 646) over established TCP sessions 645 of the network. As described above, TCP sessions are established through the IP network 605 in accordance with conventional TCP/IP transport mechanisms within the APPN/DLSw node 600 and the DLSw node 650; illustratively, these nodes cooperate in a peer-relationship to establish multiple, parallel TCP sessions 732–738 over the network.

The tagging mechanism of the commonly-assigned application incorporated by reference herein allows the hybrid node 600 to convey a TP level from its APPN layer 612 to its DLSw layer 614, convert that TP level to information that is "tagged" to each outbound packet, and allocate the tagged packet to a TCP session based on the assigned TP level. Specifically, the DLSw layer 614 loads the packets into the queue 725 and then distributes them among four queues 712–718. Each TCP session (and queue) is preferably associated with a TP level; for example, session 732 (and queue 712) are assigned a high-priority level, session 734 (and queue 714) are assigned a medium-priority level, session 736 (and queue 716) are assigned a normal-priority level and session 738 (and queue 718) are assigned a low-priority level.

The tagged information is encapsulated within an IP header of the packet prior to outbound transmission over the IP network 605 to the DLSw node 650. As noted, the tagged information is not discernible to the DLSw node 650 and, if required to respond to the packet, that node cannot select, on the basis of priority, the proper TCP session over which to transmit a corresponding inbound packet because it has no knowledge of the TP level associated with the outbound packet.

In accordance with the present invention, a mechanism is provided for conveying the TP level of an inbound packet transmitted over a heterogeneous network from DLSw node 650 to hybrid node 600. Referring to FIG. 8, the mechanism comprises a packet-recognizing filter 800 having a novel format that is generated by the hybrid node 600 and dynamically transmitted to DLSw node 650 over a predefined communication channel 850 of IP network 605. As described further herein, the filter 800 enables the switching node 650 to classify each inbound packet 810 and assign it an appropriate TP level.

According to an aspect of the invention, the predefined communication channel 800 may be implemented as either an in-band channel over one of the existing TCP sessions 732–738 using novel extensions to SSP, or an out-band channel over a newly-created TCP session. For this latter channel implementation, the newly-created TCP session is established in accordance with the conventional TCP transport protocol described above.

In another aspect of the invention, the format of filter 800 is preferably customized for each channel implementation, as depicted in FIGS. 9A and 9B. For each case, the filter includes a well-defined, unique opcode identifying the filter used in the illustrative network configuration, a format identifier (FID) denoting the format of a specific inbound packet, a local form session identifier (LFSID) that classifies the LU—LU session context of the specific packet and a priority identifier specifying the TP level of the packet.

FIG. 9A illustrates the format 900 of the filter 800 configured for transfer over the in-band communication channel. Here, the opcode 904 is encapsulated by DLSw layer 614 within a SSP header 910 along with the DLSw circuit ID 902; since the circuit ID 902 associates an end-to-end DLSw circuit with the LU—LU session of the specific inbound packet, additional addressing information is not needed. Such additional addressing information comprises media access control (MAC) addresses of the source node (SMAC) and destination node (DMAC) which, for this example, are nodes 600 and 650, respectively, and service access points (SAP) addresses of the source (SSAP) and destination (DSAP) nodes.

The format 900 further stores the FID, LFSID and priority identifiers within fields 912–916, respectively. The contents of these identifiers (along with the additional addressing information) are provided by the APPN layer 612 (FIG. 8) to the DLSw layer 614 via an application programming

interface (API) layer **820** using a data control flow mechanism, such as an API call **825**.

FIG. 9B illustrates the format **950** of the filter **800** configured for transfer over the out-of-band channel embodiment. Since a new TCP session is created for this channel embodiment, the additional addressing information passed from the APPN layer **612** to the DLSw layer **614** via the API call **825** is used in format **950**. Specifically, a defined header **970** is generated by DLSw layer **614** for encapsulating the opcode in field **954**, SMAC in field **956**, DMAC in field **958**, SSAP in field **960** and DSAP in field **962**; a value specifying the length of the filter is stored in field **952** of the header **970**. The format **950** further accomodates the FID, LFSID and priority identifiers within fields **972–976**, respectively.

Operation of the present inventive filter mechanism will now be described in connection with the flowchart of FIG. **10**. The operation starts at Step **1000** and proceeds to Step **1010** where APPN protocol layer **612** of hybrid node **600** passes filter-identifying information, such as the opcode, LFSID, FID and priority identifier, to DLSw protocol layer **614** through API layer **820** during establishment of the LU—LU session. In response to the API, the DLSw layer encapsulates these identifiers within fields of the filter as described above and transfers the filter over the communication channel **850** to the DLSw switching node **650** in Step **1015**.

Upon receiving the filter, DLSw layer **664** of the switching node **650** interprets the opcode as describing the type of message as a filter, parses the fields of the filter and stores the LFSID, FID and priority identifier in a temporary storage location **805** of, e.g., the memory of node **650** (Step **1020**). The layer **664** then proceeds to examine each inbound packet **810** prior to forwarding it to the APPN node **600**. The inbound packets are typically Systems Network Architecture (SNA) type frames generated by SNA devices coupled to the DLSw network; these frames are, in turn, typically encapsulated with conventional SNA transmission header (TH) headers. FIG. **11** is a schematic block diagram of the format **1100** of the TH header that DLSw layer **664** of node **650** is configured to operate on using the packet-recognizing filter **800**.

Specifically, the switching node determines the format of each inbound packet by initially examining the contents of a FID type field **1110** of the TH header and comparing them with the stored FID identifier in Step **1025**. It should be noted that the switching node is configured to recognize the format of TH header and, thus, can access the contents of any particular fields contained therein. The SNA architecture defines several different types of packet formats; in the illustrative embodiment, a FID2 format is preferably used for communication among the SNA devices **602,608**. If the contents of the FID type field do not match the contents of the stored FID (Step **1030**), a default priority is assigned to the inbound packet by the switching node (Step **1035**).

If there is a match in Step **1030**, the node then accesses the contents of the LFSID field **1120** of the header and compares those contents with the contents of the stored LFSID to identify the LU—LU session context of the packet (Step **1040**). If the values of the LFSIDs do not match (Step **1045**), a default priority is assigned in Step **1046**; otherwise, the switching node assigns to the inbound packet the TP level specified by the contents of the stored priority identifier in Step **1050**. That is, the DLSw layer **664** maps the packet to a selected TCP session **732–738** (FIG. **7**) based on the TP level specified by the priority identifier and loads the packet onto a corresponding queue **772–778** associated with the selected session. TCP/IP driver code within layer **666** of node **650** (FIG. **6**) then maps the TP designation of the packet to a predetermined value of precedence bits and configures those bits as a "tag" within a type of service (TOS) field when building an IP header for the packets during TCP session establishment. Thereafter, the prioritized packet is transferred through the core IP backbone network **605** to the hybrid node **600** in Step **1055** and the operation ends in Step **1060**.

Advantageously, the present invention enables the APPN layer of a hybrid node to instruct the DLSw layer of a receiving switching-node as to the TP level of an inbound packet destined for the hybrid node. In response to the instruction, the switching node assigns the inbound packet to a particular TCP session, where priority is preserved at intermediate queuing points on the basis of the value of the precedence bits in the IP header.

While there has been shown and described an illustrative embodiment for conveying information pertaining to priority levels of inbound packets transmitted over a heterogeneous network from a switching node to a hybrid node of the network, it is to be understood that various other adaptations and modifications may be made within the spirit and scope of the invention. For example in an alternate embodiment of the invention, a different transport mechanism may be employed in the heterogeneous network to transport different packet formats among the end nodes, such as NetBIOS devices, of the network. Here, another unique, yet well-defined opcode is needed to identify the filter used to convey the priority levels of these different inbound packets over the alternately-configured network.

The foregoing description has been directed to specific embodiments of this invention. It will be apparent, however, that other variations and modifications may be made to the described embodiments, with the attainment of some or all of their advantages. Therefore, it is the object of the appended claims to cover all such variations and modifications as come within the true spirit and scope of the invention.

What is claimed is:

1. Apparatus for conveying information pertaining to transmission priority (TP) levels of inbound packets transmitted over a heterogeneous network from a switching node, the apparatus comprising:

a hybrid node for being coupled to a predefined communication channel for interconnecting the hybrid node and the switching node, the hybrid node being configured to generate a packet-recognizing filter for being transmitted to the switching node over the predefined communication channel, the filter enabling the switching node to classify the inbound packets and assign them appropriate TP levels.

2. An apparatus according to claim **1**, wherein the predefined communication channel is one of an in-bound channel over an existing transport session connection between the nodes and an out-of-band channel over a newly-created transport session.

3. An apparatus according to claim **2**, wherein the filter comprises identifiers identifying attributes of the inbound packets, such that inbound packets matching these identifiers are associated with appropriate TP levels.

4. An apparatus according to claim **3**, wherein the identifiers comprise a local form session identifier classifying the session context of a specific inbound packet and a priority identifier specifying the TP level of the packet.

5. An apparatus according to claim **4**, wherein the hybrid node comprises an Advanced Peer to Peer Networking

(APPN) protocol layer and a Data Link Switching (DLSw) protocol layer, and wherein the APPN protocol layer passes the identifiers to the DLSw protocol layer of the hybrid node through an application programming interface.

6. An apparatus according to claim **5**, wherein the DLSw protocol layer encapsulates the identifiers within fields of the filter and transfers the filter over the predefined communication channel.

7. An apparatus according to claim **6**, wherein the opcode is encapsulated within a switch-to-switch protocol header when transferring the filter over the in-band channel.

8. An apparatus according to claim **6**, wherein the opcode and additional addressing information are encapsulated within a defined header when transferring the filter over the out-of-band channel.

9. An apparatus according to claim **1**, wherein the hybrid node comprises an Advanced Peer to Peer Networking (APPN) protocol layer and a Data Link Switching (DLSw) protocol layer, and wherein the filter comprises identifiers including a unique opcode identifying the filter, a format identifier (FID) denoting the format of a specific inbound packet, a local form session identifier (LFSID) classifying the session context of the specific inbound packet and a priority identifier specifying a TP level of the packet.

10. An apparatus according to claim **9**, wherein the identifiers are for being passed from the APPN protocol layer to the DLSw protocol layer of the hybrid node through an application programming interface (API).

11. An apparatus according to claim **10**, wherein the identifiers are encapsulated within fields of the filter at the DLSw layer in response to the API.

12. An apparatus according to claim **11**, wherein the predefined communication channel is an in-band channel over one of a plurality of existing transport session connections between the nodes or an out-band channel over a newly-created transport session between the nodes.

13. An apparatus according to claim **12**, wherein the opcode is encapsulated within a switch-to-switch protocol header.

14. An apparatus according to claim **12**, wherein opcode and additional addressing information are encapsulated within fields of a defined header.

* * * * *

US006385207B1

(12) **United States Patent**
Woundy

(10) **Patent No.: US 6,385,207 B1**
(45) **Date of Patent: *May 7, 2002**

(54) **RSVP SUPPORT FOR UPSTREAM TRAFFIC**

(75) Inventor: **Richard Woundy**, North Reading, MA (US)

(73) Assignees: **MediaOne Group, Inc.**, Englewood, CO (US); **U S West, Inc.**, Denver, CO (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **09/468,032**

(22) Filed: **Dec. 20, 1999**

Related U.S. Application Data

(63) Continuation of application No. 08/996,349, filed on Dec. 23, 1997, now Pat. No. 6,031,841.

(51) **Int. Cl.**[7] **H04L 12/28**
(52) **U.S. Cl.** **370/410**; 370/443; 370/468
(58) **Field of Search** 370/351, 352, 370/353, 356, 384, 385, 395, 401, 410, 414, 431, 437, 443, 447, 465, 468, 480, 485, 486, 487; 375/222

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,574,724 A		11/1996	Bales et al.	
5,592,477 A		1/1997	Farris et al.	
5,621,728 A		4/1997	Lightfoot et al.	
5,631,903 A		5/1997	Dianda et al.	
5,648,958 A	*	7/1997	Counterman	370/458
5,677,905 A		10/1997	Bigham et al.	
5,734,833 A	*	3/1998	Chiu	395/200.55
5,963,557 A	*	10/1999	Eng	370/432
6,031,841 A	*	2/2000	Woundy	370/410
6,031,844 A	*	2/2000	Lin	370/431

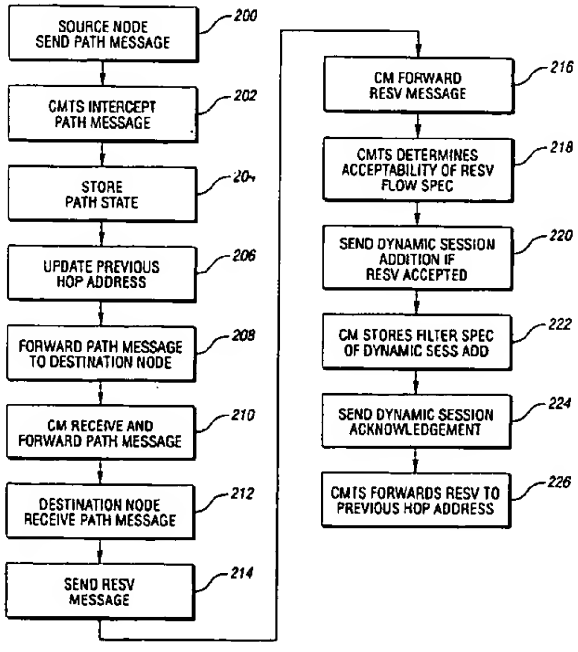
* cited by examiner

Primary Examiner—Ricky Ngo
(74) *Attorney, Agent, or Firm*—Brooks & Kushman P.C.

(57) **ABSTRACT**

A method for managing MAC messages is provided which supports dynamic resource reservation for upstream data traffic in a broadband cable system. Three specialized MAC management messages of Dynamic Session Addition, Dynamic Session Deletion, and Dynamic Session Acknowledgment are used to control setting of a filter spec parameter in a cable modem. The present invention provides upstream bandwidth resource reservation which allows packet scheduling to occur at a CMTS, and packet classification to occur at a cable modem. Such an arrangement allows the RSVP protocol to be adapted for use at the OSI protocol layer of a typical cable modem, thereby improving efficiency of network bandwidth/resource allocation.

6 Claims, 1 Drawing Sheet



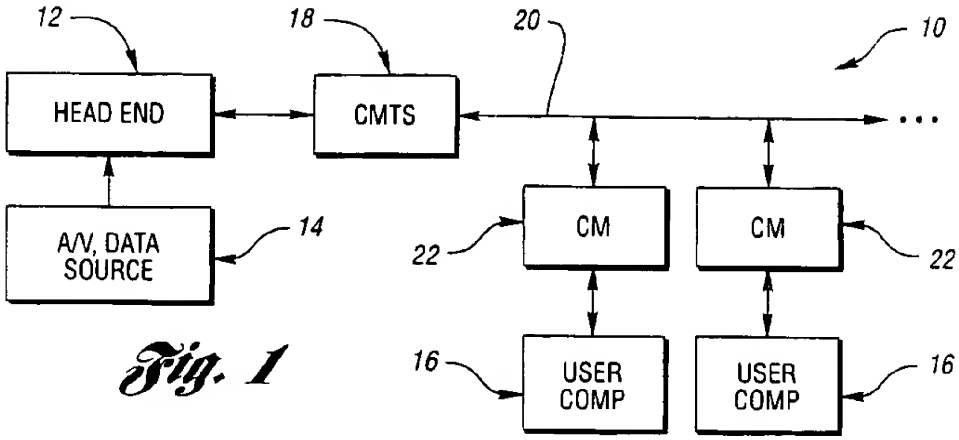

12
18
20
10
HEAD END
CMTS
A/V, DATA SOURCE
14
22
CM
CM
22
16
USER COMP
USER COMP
16

Fig. 1

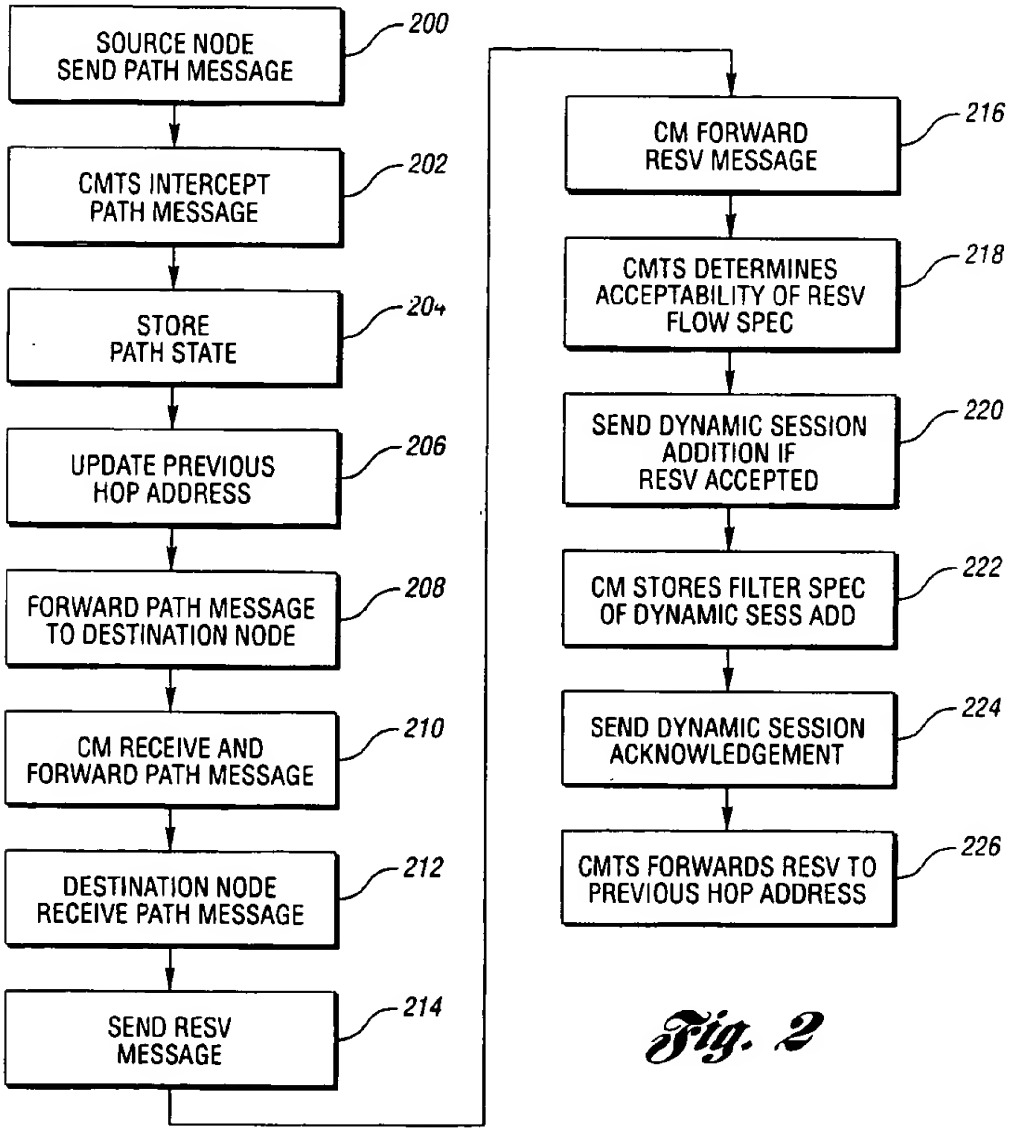

SOURCE NODE SEND PATH MESSAGE
200
CMTS INTERCEPT PATH MESSAGE
202
STORE PATH STATE
204
UPDATE PREVIOUS HOP ADDRESS
206
FORWARD PATH MESSAGE TO DESTINATION NODE
208
CM RECEIVE AND FORWARD PATH MESSAGE
210
DESTINATION NODE RECEIVE PATH MESSAGE
212
SEND RESV MESSAGE
214
CM FORWARD RESV MESSAGE
216
CMTS DETERMINES ACCEPTABILITY OF RESV FLOW SPEC
218
SEND DYNAMIC SESSION ADDITION IF RESV ACCEPTED
220
CM STORES FILTER SPEC OF DYNAMIC SESS ADD
222
SEND DYNAMIC SESSION ACKNOWLEDGEMENT
224
CMTS FORWARDS RESV TO PREVIOUS HOP ADDRESS
226

Fig. 2

RSVP SUPPORT FOR UPSTREAM TRAFFIC

This application is a continuation of application Ser. No. 08/996,349 filed on Dec. 23, 1997, now U.S. Pat. No. 6,031,841.

TECHNICAL FIELD

The present invention generally relates to systems for providing internetwork data distribution over a broadband cable data network, and more particularly to a method and system for MAC message management to support dynamic bandwidth/resource reservation for upstream traffic in a data-over-cable context.

BACKGROUND ART

Resource ReSerVation Protocol (RSVP) is a resource reservation setup protocol currently being standardized by the Internet Engineering Task Force (IETF). RSVP provides receiver-initiated setup of resource reservations for multicast and unicast data flows.

Referring to FIG. **1**, a basic broadband cable data distribution system **10** is shown as having a cable head end/distribution hub **12** connected to one or more audio/video and data sources **14**, such as a satellite receiver. The cable head end **12** distributes the received signals to a plurality of end user computers **16** using at least one cable modem termination service (CMTS) **18** and a coaxial or hybrid optical/coaxial cable **20**. The CMTS **18** provides dynamic allocation or reservation of network bandwidth/resources to selectively control access and quality of service to the network for end users **16**. The end users are typically connected to the network via a cable modem (CM) **22**. With respect to downstream bandwidth, end users share the bandwidth in accordance with a time sharing allocation protocol, such as an Ethernet contention protocol, or a specially granted service I.D. (SID) generated by the CMTS in accordance with a predetermined time allocation map.

Generally, the basic RSVP protocol assumes the implementation of two modules on each RSVP-capable node to forward data packets: the "packet classifier" and the "packet scheduler." The packet classifier determines the route and quality of service (QOS) class for each packet, and sends the packet to the packet scheduler. The RSVP packet classifier uses a "filter spec" which matches a particular source internet protocol (IP) address and source port to classify and restrict traffic that consumes reservation resources/bandwidth. The packet scheduler makes packet forwarding decisions such as queuing decisions to achieve a predetermined QOS on the interface. The RSVP packet scheduler uses a "flow spec" which identifies token packet parameters, peak data rate, etc. to identify the desired QOS.

In the context of RSVP for upstream traffic, it is desirable for the CM to perform the "packet classifier" function, but the CMTS to perform most of the "packet scheduler" function. While CMTS have utilized different levels of SIDs relating to different levels of quality of services, such arrangements have been fixed and inefficient in that there was no ability to control how much of the available resources a user would be able to reserve. Thus, a need exists for an arrangement which can adequately support a split of function between the CMTS and the CM to dynamically allow reservation of available upstream bandwidth and corresponding improvement in system efficiency.

DISCLOSURE OF THE INVENTION

It is therefore an object of the present invention to provide a method for managing MAC messages to support resource reservation for upstream traffic in a broadband cable system, i.e., a data-over-cable context.

In accordance with this and other objects, the present invention supports the above-noted split of function by providing a method for MAC management which utilizes three specialized MAC management messages of Dynamic Session Addition, Dynamic Session Deletion, and Dynamic Session Acknowledgment, and a management protocol relating thereto.

More specifically, the present invention provides a method for dynamically allocating upstream network resources in a broadband cable data distribution network having the steps of transmitting a resource reservation signal from a cable modem indicative of an amount of network resources needed for upstream communication, and in response to the reservation signal, determining at a CMTS whether upstream resources are available. If so, a SID message is generated at the CMTS to indicate at least one filter spec parameter responsive to the reserved resources. Once the SID message is received at the cable modem, a filter spec parameter is set in the cable modem equal to the at least one filter spec parameter in the received SID message. Then, an upstream communication matching the set filter spec can be sent by the cable modem.

The above objects and other objects, features, and advantages of the present invention are readily apparent from the following detailed description of the best mode for carrying out the invention when taken in connection with the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is a block diagram of a basic cable data network system; and

FIG. **2** is a flowchart illustrating resource reservation and MAC management operation in accordance with the present invention.

BEST MODE FOR CARRYING OUT THE INVENTION

The present invention provides a MAC message management arrangement which allows dynamic reservation of upstream traffic by a cable modem (CM), and can operate in conjunction and support of RSVP. More specifically, a Dynamic Session Addition message is periodically transmitted from a CMTS to a CM to announce the allocation of a new SID. The Dynanic Session Addition message contains a new SID value, and type/length/value fields which can encode a RSVP filter spec and RSVP "cleanup timeout" interval to support the RSVP "soft state" approach. The CM is arranged to use the new SID exclusively for upstream traffic that matches the filter spec, thereby allowing classification to be performed by the CM. The CM preferably assumes that a SID is refreshed by the receipt of another Dynamic Session Addition message within the cleanup timeout interval. Otherwise, the SID is ignored by the CM at the conclusion of the interval.

In addition, a Dynamic Session Deletion message can be transmitted from the CMTS to the CM to delete an unused SID immediately, thereby supporting an RSVP explicit "teardown" message. A Dynamic Session Acknowledgment message is transmitted from the CM to the CMTS to acknowledge receipt of a Dynamic Session Addition or Dynamic Session Deletion message.

Referring now to FIG. **2**, a flowchart illustrates overall management between an explicit RSVP "Path" and "Resv"

message with the Dynamic Session messages of the present invention. More specifically, a data flow source-node generates an RSVP Path message and sends the message toward a data flow destination-node at block **200**. The CMTS will intercept the downstream RSVP Path message at block **202**, store the path state from the message at block **204**, update the previous hop address in the message at block **206** to include the CMTS address, and forward the message at block **208**. As shown at block **210**, the CM then forwards the downstream RSVP Path message to the destination-node without processing.

At block **212**, the data flow destination-node receives the RSVP Path message, and replies at block **214** with an RSVP Resv message to request an upstream reservation of bandwidth resources for the data flow to be sent by the source-node to the destination node. The RSVP Resv message is sent to the previous hop address of the Path message, i.e., the CMTS. At block **216**, the CM forwards the upstream RSVP Resv message to the CMTS without processing. When the CMTS receives the upstream RSVP Resv message, the message flow spec is processed at block **218** using an admission control and policy control protocol in cooperation with the CMTS upstream bandwidth scheduler.

Upon acceptance of the Resv reservation message by the CMTS, the CMTS then sends a Dynamic Session Addition MAC message as described above to the CM at block **220**. The message includes a new SID and the filter spec from the RSVP Resv message. The CM receives the Dynamic Session Addition MAC message, stores the new SID and filter spec at block **222**, and sends a Dynamic Session Acknowledgment MAC message back to the CMTS at block **224**. The CMTS receives the Dynamic Session Acknowledgment MAC message at block **224**, and forwards the RSVP Resv message to the previous hop address at block **226**. Communication can then take place.

Thus, the present invention provides upstream bandwidth resource reservation which allows packet scheduling to occur at the CMTS, and packet classification to occur at a CM. Such an arrangement allows the RSVP protocol to be adapted for use at the OSI protocol layer of a typical CM, thereby improving efficiency of network resource allocation.

While the best mode for carrying out the invention has been described in detail, those familiar with the art to which this invention relates will recognize various alternative designs and embodiments for practicing the invention as defined by the following claims.

What is claimed is:

1. A method for dynamically allocating upstream network resources in a packet-based broadband cable data distribution network using RSVP protocol and having a CMTS (cable modem termination system) connected to each of a plurality of user terminals via a cable modem, said method comprising:

generating an RSVP protocol path message at a source node for establishing communication with a destination node;

passing the RSVP path message to the destination node;

performing packet classification at a destination node cable modem by transmitting a resource reservation signal based on the received RSVP path message to request an amount of network resources needed to be reserved for upstream communication with the source node;

performing packet scheduling at a CMTS by determining whether network resources are available based on the reservation signal, and if so, generating at the CMTS a SID (service identification) message comprising at least one filter spec parameter responsive to the reserved network resources;

receiving the SID message at the destination node cable modem; and

setting a filter spec parameter in the destination node cable modem equal to the at least one filter spec parameter in the received SID message for sending an upstream communication matching the set filter spec parameter.

2. The method of claim **1** further comprising sending an acknowledgment signal from the cable modem to the CMTS after setting of the filter spec parameter.

3. The method of claim **1** further comprising detecting that a SID has not been utilized by a cable modem, and sending a reservation termination message from the CMTS indicating termination of the SID message.

4. The method of claim **1** wherein passing the RSVP path message to the destination node comprises updating a hop address in the message to include a CMTS address, and sending the resource reservation signal to the CMTS using the updated hop address.

5. The method of claim **1** wherein the CMTS performs the packet scheduling by processing a flow spec parameter in the received resource reservation signal.

6. The method of claim **1** wherein the SID is sent as part of a dynamic session addition MAC message.

* * * * *